



**UNIVERSIDAD CATÓLICA ANDRÉS BELLO**  
VICERRECTORADO ACADÉMICO  
ESTUDIOS DE POSTGRADO  
ÁREA DE INGENIERÍA  
Postgrado en Sistemas de Información

**Trabajo de Grado de Maestría**

MODELO PARA LA SELECCIÓN DE LA METODOLOGÍA DE DESARROLLO  
WEB DE UNA APLICACIÓN SEGÚN SUS CARACTERÍSTICAS FUNCIONALES

Presentado por  
Vilariño De Almeida Julio Cesar;  
Para optar al título de  
Magister en Sistemas de Información

Tutor  
Bianchini Adelaide

Caracas, 19 de Mayo del año 2010

## Tabla de Contenido.

<b>Resumen</b> .....	<b>i</b>
<b>Introducción</b> .....	<b>1</b>
I.1 Objetivo General.....	7
I.1.1 Objetivos específicos.....	7
<b>CAPÍTULO II: Marco Teórico</b> .....	<b>9</b>
II.1 Conceptos Preliminares.....	10
II.2 Modelos de referencia y modelos de hipermedia .....	11
II.3 Tipos de aplicaciones Web.....	12
II.4 Marco de referencia para el estudio de las metodologías.....	17
II.4.1 Modelos fundamentales en el desarrollo de aplicaciones Web.....	19
II.5 Metodologías de desarrollo Web.....	24
II.5.1 HDM – Hypermedia Design Model .....	25
II.5.2 RMM - Relationship Management Methodology .....	28
II.5.3 OO-H – Object-Oriented Hypermedia.....	30
II.5.4 EORM – Enhanced Object Relationship Methodology .....	33
II.5.5 OOHDM - Object-Oriented Hypermedia Design Method .....	34
II.5.6 SOHDM – Scenario-based Object-Oriented Hypermedia Design Methodology .....	36
II.5.7 WebML – Web Modeling Language.....	39
II.5.8 W2000 .....	40
II.5.9 UWE – UML-based Web Engineering.....	42
II.5.10 WSDM – Web Services Distributed Management .....	43
II.5.11 OOWS – Object Oriented Web Solutions .....	45
II.5.12 NDT – Navigational Development Techniques .....	46
II.6 Herramientas CASE .....	50
II.7 Criterios relacionados a los equipo de trabajo que serán usados en el modelo.....	54
II.7.1 Experiencia en desarrollo Web .....	54
II.7.2 Experiencia en modelos de procesos .....	54

II.7.3 Experiencia en paradigma de modelado.....	56
II.7.4 Experiencia en lenguaje de modelado .....	57
<b>CAPÍTULO III: Marco Metodológico.....</b>	<b>58</b>
<b>CAPÍTULO IV: Modelo de selección de la mejor metodología de desarrollo para una aplicación Web.....</b>	<b>62</b>
IV.1 Relación de las fases del proceso de los métodos de modelado .....	63
IV.2 Tipos de aplicaciones Vs. tipos de modelado requerido .....	67
IV.3 Tipos de modelado que componen cada metodología de diseño Web ...	71
IV.4 Reducción de los métodos de modelado Web. ....	73
IV.5 Análisis de las herramientas CASE de las metodologías de diseño Web .....	75
IV.6 Matriz de selección de la metodología de desarrollo Web. ....	76
IV.7 Prácticas en desarrollo Web.....	79
IV.8 Modelo propuesto para la selección de la metodología de desarrollo Web de una aplicación según sus características funcionales .....	81
<b>CAPÍTULO V: Ejemplos del funcionamiento del modelo de selección de la mejor metodología de desarrollo para una aplicación Web.....</b>	<b>85</b>
V.1 Ejemplo N° 1. ....	85
V.2 Ejemplo N° 2. ....	89
V.3 Ejemplo N° 3. ....	92
<b>CAPÍTULO VI: Conclusiones y recomendaciones .....</b>	<b>97</b>
<b>CAPÍTULO VII: Referencias Bibliográficas. ....</b>	<b>102</b>
<b>APÉNDICE 1 .....</b>	<b>108</b>

## Índice de Figuras.

Figura N° 1. Cantidad de dominios registrados entre diciembre 2000 y septiembre 2009. Tomado de (Zakon, 2010) .....	1
Figura N° 2. Ambiente de desarrollo genérico (Bianchini, 2000).....	10
Figura N° 3. Características a considerar en el desarrollo de aplicaciones Web. .....	17
Figura N° 4. Esquema HDM. Tomado de Garzotto (Garzotto, Paolini, & Schwabe, 1993).....	27
Figura N° 5. Diagrama RMDM. Tomado de (Isakowitz, Stohr, & Balasubramanian, 1995).....	30
Figura N° 6. Modelo para la selección de la metodología de desarrollo Web de una aplicación según sus características funcionales.....	81

## Índice de Tablas.

Tabla 1. Comparativa de los aspectos relevantes de las metodologías de desarrollo Web.....	49
Tabla 2. Comparaciones de las herramientas CASE de las diferentes metodologías de desarrollo Web estudiadas .....	53
Tabla 3. Modelos de procesos .....	55
Tabla 4. Actividades/resultados por cada etapa planificada para la ejecución de la investigación .....	60
Tabla 5. Comparación de las fases de las metodologías de desarrollo Web con el ciclo de vida .....	63
Tabla 6. Tipo de aplicaciones Vs. tipo de modelados .....	70
Tabla 7. Tipos de modelados que contiene cada Metodologías de desarrollo Web estudiada en este trabajo de grado.....	71
Tabla 8. Leyenda correspondiente a la Tabla 9.....	76
Tabla 9. Matriz de selección de la metodología de desarrollo Web, según el tipo de aplicación a realizar. ....	77
Tabla 10. Solución del cuestionario para el ejemplo N°1 .....	86
Tabla 11. Resultados de la aplicación del modelo de selección para el ejemplo N° 1 .....	88
Tabla 12. Solución del cuestionario para el ejemplo N°2 .....	90
Tabla 13. Resultados de la aplicación del modelo de selección para el ejemplo N° 2.....	92
Tabla 14. Solución del cuestionario para el ejemplo N°3 .....	94
Tabla 15. Resultados de la aplicación del modelo de selección para el ejemplo N° 3.....	96

## Resumen

El volumen de aplicaciones desarrolladas en ambiente Web, ha tenido un crecimiento exponencial en las últimas décadas. Esto ha generado un gran inconveniente dentro de los equipos de desarrollo de aplicaciones, debido a que en un principio han querido utilizar metodologías de desarrollo de software tradicionales en la construcción de aplicaciones Web. Esto ha sido una estrategia equivocada por parte de los equipos de trabajo, porque las aplicaciones Web poseen características especiales, ya que están compuestas por nodos y los usuarios tienen la libertad de navegar entre ellos, lo que las metodologías tradicionales no consideran, produciendo un diseño inadecuado de la aplicación.

En la actualidad existe una variedad de metodologías diseñadas para el desarrollo de aplicaciones Web. En su mayoría son metodologías prescriptivas, utilizadas para cualquier circunstancia de desarrollo.

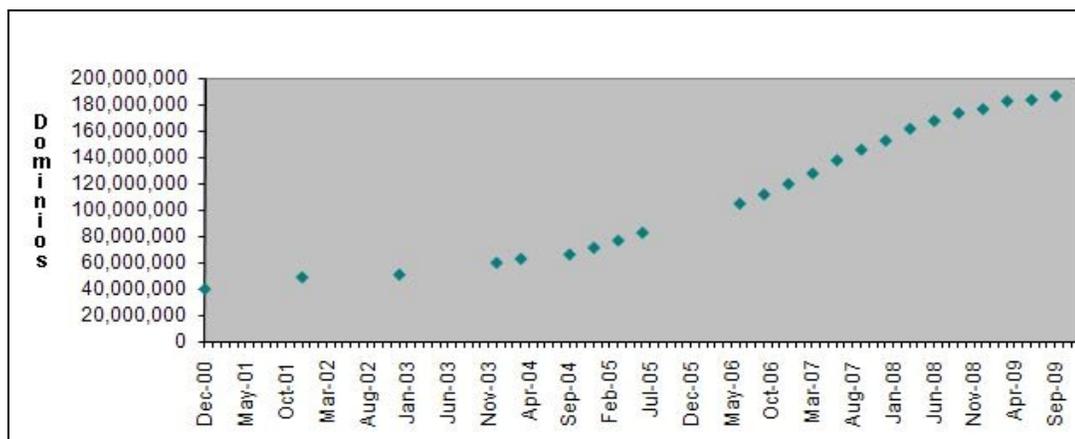
El objetivo de este trabajo es el desarrollo de un modelo que guíe a un equipo desarrollador en la selección de la metodología más adecuada para el desarrollo de una aplicación Web. Para esto se consideraron algunos aspectos relacionados con el entorno de desarrollo, el producto final, las herramientas disponibles, y la experiencia de los desarrolladores.

También como parte de los objetivos de esta investigación está el comparar las principales metodologías de desarrollo de aplicaciones Web, definir

los tipos de aplicaciones y los artefactos necesario para su diseño según las funciones que va a desempeñar e identificar los aspectos más relevantes relacionados con las características del equipo de desarrollo, que deben tomarse en cuenta a la hora de seleccionar la metodología de desarrollo.

## Introducción.

El crecimiento exponencial de la cantidad de aplicaciones Web, como se puede observar en la figura N°1 obtenida de (Zakon, 2010), y la importancia que han tomado en el campo empresarial y comercial, ha hecho que cada vez más empresas y organizaciones se vean atraídas por tener sus servicios disponibles en la plataforma Web. Las aplicaciones Web poseen características particulares diferentes a las aplicaciones tradicionales, por lo tanto, las metodologías de desarrollo de software convencionales no cumplen con las herramientas necesarias para el diseño de aplicaciones Web. Si se utiliza una metodología de desarrollo de aplicaciones tradicionales para llevar a cabo una aplicación Web, el diseño de ésta no será adecuado y la probabilidad de un fracaso, en distintas instancias del proyecto, pudiese ser muy elevada.



**Figura N° 1.** Cantidad de dominios registrados entre diciembre 2000 y septiembre 2009.  
Tomado de (Zakon, 2010)

Por lo anterior, en la última década se han propuesto diferentes metodologías de desarrollo Web, las cuales cuentan con las herramientas requeridas para el diseño de este tipo de aplicación, ayudando así a los desarrolladores a diseñar aplicaciones efectivas que cumplan los objetivos para las cuales fueron construidas.

En este trabajo de grado se busca generar una herramienta que ayude a los equipos de trabajo a seleccionar la metodología de desarrollo Web más adecuadas para llevar a cabo un proyecto, según sus características funcionales, como la complejidad de la aplicación que se vaya a realizar y la experiencia que posee el equipo en aspectos como: lenguaje de notación, modelado de proceso, entre otros.

El trabajo de grado consta de los siguientes capítulos: en el capítulo I se expone el planteamiento del problema y los objetivos que persigue este trabajo de grado; en el capítulo II se expone el marco teórico de la investigación; en el capítulo III se presenta el marco metodológico que se sigue para la realización de la investigación; en el capítulo IV se presenta la formulación del modelo; en el capítulo V se presentan tres ejemplos donde se muestra el funcionamiento del modelo. En el capítulo VI se presentan las conclusiones y los resultados y por último se presentan las referencias

bibliográficas consultadas para la revisión sobre el estado del arte y la información necesaria para llevar a cabo este trabajo de grado.

## **CAPÍTULO I: Planteamiento del problema.**

El desarrollo de aplicaciones Web ha ido aumentando de manera considerable y ha tomado gran importancia dentro de las organizaciones. Pero muchas de estas aplicaciones no han tenido el éxito esperado, ya que no cumplen con los objetivos para los cuales fueron desarrolladas.

Un estudio realizado por el consorcio consultor Cutter Mike EPNER (Cutter, 2000), muestra los graves inconvenientes que han venido arrastrando las aplicaciones Web. A continuación se presentan parte de los resultados obtenidos en el estudio:

- El 79% de los proyectos de desarrollo de aplicaciones Web no cumplió el tiempo establecido para su ejecución.
- El 63% de los proyectos tuvo un presupuesto superior a lo planificado
- El 53% de las aplicaciones Web no cumple con los requerimientos funcionales mínimos.
- El 52% de las aplicaciones Web son de baja calidad.

El problema de un mal diseño en una aplicación Web se produce principalmente por dos causas: la falta de conocimiento del entorno Web y por la utilización de metodologías no adecuadas para el desarrollo de este tipo de aplicaciones. Las aplicaciones Web, a diferencia de las tradicionales deben cumplir con estándares de usabilidad por la capacidad de interacción

con el usuario y los aspectos de la presentación de la información, de forma que no genere carga cognitiva (dificultad de la aplicación para ser utilizada), y no haya desorientación del usuario debido a el modelado de las navegaciones en el ambiente hipertexto sobre el cual se implementan tales aplicaciones. (Bianchini, 2000).

Los desarrollos en ambientes Web llevados a cabo hacia finales de los años 90 utilizaban adecuaciones de metodologías para desarrollo de software tradicional. Es a principios de este siglo que las propuestas metodológicas en el ámbito de desarrollo Web han madurado y se ha constituido la disciplina de la Ingeniería Web (Deshpande, Murugesan, Hansen & Ginige, 2001) quien le da forma y genera el primer concepto sobre la disciplina, enunciándola como: “la aplicación de enfoques sistemáticos y cuantificables (conceptos, métodos, técnicas y herramientas) para llevar a cabo el análisis de requerimientos, diseño, implementación, pruebas, operación y mantenimiento de aplicaciones Web de alta calidad”.

A lo largo de los últimos 10 años se han creados metodologías de desarrollo Web que permiten mejorar el análisis y diseño de los sistemas realizados bajo esta tecnología. De ahí que hayan disminuido de manera considerable los problemas de diseño de los sistemas de software Web. Las aplicaciones Web pueden alcanzar distintos niveles de complejidad y están

atadas al concepto de presencia en el World Wide Web (WWW). Se estima que una vez levantados los requerimientos principales, los clientes presionan por el desarrollo, por lo tanto los tiempos son muchos más cortos que en el caso de aplicaciones tradicionales. Retomando la característica sobre interacción, navegación y presentación de la información se puede deducir que, además, el equipo de desarrollo debe ser multidisciplinario. (Diaz & Aedo, 2007)

Por lo anterior, existen ciertos atributos relacionados con el desarrollo de aplicaciones Web que los líderes de este tipo de proyecto deben saber gestionar, tal como son: la selección de un método de desarrollo adecuado, enfoque metodológico (Ágil Vs. disciplinado), modelo de proceso, según tipo de aplicación a desarrollar su complejidad, número y competencias de personal disponible, entre otros.

En la actualidad existen estudios que han analizado esta problemática y han intentado darle una solución, como es el caso expuesto en (Selmi, Kraïem, Ghézala, & Hajjami, 2008), donde propone un meta-modelo compuesto de métodos y es el propio equipo de trabajo quien crea la metodología que va a utilizar según: el tipo de aplicación a desarrollar, la complejidad de la aplicación, similitud con otras aplicaciones desarrolladas y los problemas en la definición y claridad de los objetivos de la aplicación.

Por lo expuesto anteriormente, en este trabajo de grado se propone un modelo que servirá como guía o apoyo a los desarrolladores de aplicaciones Web a la hora de decidir que metodología de desarrollo van a utilizar para llevar a cabo el proyecto.

### **I.1 Objetivo General.**

Generar un modelo que permita seleccionar la metodología de desarrollo para una aplicación Web, según sus características funcionales y la experiencia del equipo de trabajo que realizará la misma, para disminuir los problemas de diseño y desarrollo.

#### **I.1.1 Objetivos específicos.**

- Analizar y comparar las principales metodologías de desarrollo de aplicaciones Web existentes.
- Definir las características de una aplicación Web que deben tomarse en cuenta a la hora de seleccionar la metodología de desarrollo que se va a utilizar.
- Definir los aspectos más relevantes relacionados con las características del equipo de desarrollo que deben tomarse en cuenta a la hora de seleccionar la metodología a utilizar.

- Definir y diseñar un modelo que dadas las características de la aplicación web y del equipo de trabajo, seleccione la metodología de desarrollo más adecuada para su análisis y diseño.

## **CAPÍTULO II: Marco Teórico.**

En este capítulo se exponen los fundamentos teóricos que fueron necesarios para la construcción del modelo de selección de la metodología para el diseño de una aplicación Web. Se describirán conceptos generales, pero el énfasis principal será en aquellas descripciones asociadas a las metodologías que se han evaluado para realizar este trabajo de grado. Los contenidos que se discutirán se basan en el esquema que se muestra en la figura N°2, que representa un ambiente de desarrollo genérico, donde se identifican:

- a. Una metodología constituida por: actividades, roles y artefactos, siguiendo algún modelo de proceso.
- b. Herramientas que pueden dar soporte al proceso metodológico, con los cuales se pueden gestionar actividades y generar artefactos los cuales representan modelos descritos con algún lenguaje de modelado y siguiendo algún paradigma establecido.
- c. Equipo de desarrollo, que son personas y competencias que llevan a cabo uno o más roles dentro del enfoque metodológico.

Por lo expuesto, el capítulo se estructura de la siguiente manera: en la sección II.1 se exponen los conceptos preliminares necesarios para comprender la terminología Web. En la sección II.2 se explican los modelos

referenciales y de hipertexto. En la sección II.3 se presentan los tipos de aplicaciones Web que existen actualmente en el mercado, en la sección II.4 se describen brevemente las distintas propuestas metodológicas estudiadas en el presente trabajo, y por último en la sección II.5 se exponen los criterios relacionados a los equipos de trabajo que serán tomados en cuenta en el modelo de selección.

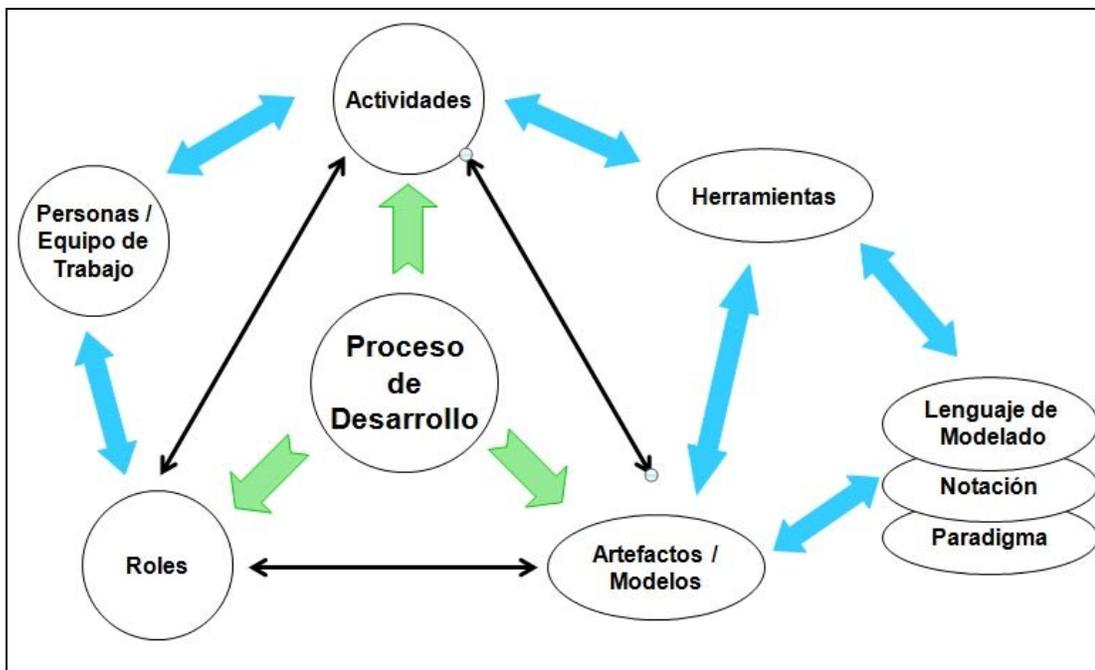


Figura N° 2. Ambiente de desarrollo genérico (Bianchini, 2000)

## II.1 Conceptos Preliminares.

**Hipertexto:** “se entiende como la organización de una base de información en bloques discretos de contenido llamado nodos (en su mínimo nivel),

conectados a través de enlaces cuya selección genera distintas formas de recuperar la información de la base” (Bianchini, 2000).

**Hipermedia:** es un hipertexto que tiene contenidos multimedia, que “consiste en la tecnología que utiliza la información almacenada en diferentes formatos y medios, controlados por un usuario” (Bianchini, 2000), como por ejemplo: videos, imágenes, sonido, entre otros.

**Hiperdocumento:** “es el contenido de información, incluyendo los fragmentos de información y las conexiones entre esos fragmentos, indiferente sea el sistema utilizado para leer o escribir tal documento” (Bianchini, 2000).

**Aplicación Web:** “es un sistema de software basado en tecnologías y estándares del World Wide Web Consortium (W3C), que provee los recursos específicos Web, tales como contenido y servicios a través de una interfaz de usuario como lo es un navegador Web.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

## **II.2 Modelos de referencia y modelos de hipermedia**

En este trabajo de grado se desarrolló un modelo para la selección de la mejor metodología de desarrollo de una aplicación Web. Es importante señalar las diferencias entre metodologías de diseño, las cuales incluyen “un conjunto de pasos y procedimientos que deben seguirse para el desarrollo de

una aplicación” (Piattini, 2007). Por lo tanto, son una serie de actividades que se llevan a cabo para establecer los objetivos y el funcionamiento de los componentes que forman parte de una aplicación Web. Por otro lado existen: i) los modelos de referencia que “captan las abstracciones más importantes que se encuentran en una amplia gama de sistemas de hipertexto existentes y futuras” (Halasz & Schwartz, 1994), y ii) los modelos de hipermedia que “aportan una base teórica y formal que permite identificar claramente los elementos, atributos y operaciones de los sistemas hipermedia”. (Pastor & Saorín, 1998). A continuación se nombran algunos de estos modelos.

- El modelo de Dexter (Halasz & Schwartz, 1994)
- El modelo de Ámsterdam (Hardman, Bulterman, & Van Rossum, 1994)
- The Hypertext Abstract Machine (HAM) (Campbell & Goodman, 1988)
- El modelo de Trellis (Furuta & Stotts, 1989)
- El modelo formal de Lange (Lange, 1990)
- El modelo Tower (De Bra, Houben, & Kornatzky, 1993)
- El modelo de Hipermedio basado en teoría de grafos (Bianchini, 2000)

### **II.3 Tipos de aplicaciones Web.**

El avance que ha tenido la tecnología Web ha generado aplicaciones que van desde páginas Web, que sólo muestran información estática a inmensos portales como tiendas virtuales y banca electrónica, entre otros.

Todo esto ha generado que las aplicaciones Web se puedan clasificar según su funcionalidad. A continuación mostraremos los tipos de aplicaciones Web según la propuesta de (Kappel, Pröll, Reich, & Retschitzegger, 2003):

- **Sitios Web centrados en la información:** “Son los precursores de las aplicaciones Web. Son páginas Web prefabricadas y almacenadas en servidores Web. Por ejemplo: páginas Web estáticas, documentos HTML que son enviados a los usuarios cuando realizan alguna petición. Estas páginas Web son modificadas manualmente o usando las herramientas respectivas.” (Kappel, Pröll, Reich, & Retschitzegger, 2003), En otras palabras, los sitios Web centrados en la información son páginas Web donde la interacción con el usuario es limitada (lectura y navegación sencilla), ya que su contenido es estático y la mayoría de las veces son modificadas de forma manual. Ejemplo: páginas Web de información general de compañías ([www.cvg.com](http://www.cvg.com)), páginas Web personales, etc.
- **Aplicaciones Web interactivas:** “Emergen ofreciendo una simple forma de interactividad a través de las páginas de un sitio Web... páginas Web son dirigidas a otras páginas dinámicamente generadas por las entradas del usuario.” (Kappel, Pröll, Reich, & Retschitzegger, 2003), es decir, son aplicaciones Web las cuales interactúan con el

usuario y donde su contenido está en constante cambio. Ejemplo: portales de noticias ([www.globovision.com](http://www.globovision.com), [www.eluniversal.com](http://www.eluniversal.com), [www.el-nacional.com](http://www.el-nacional.com)), portales donde se realizan exhibiciones virtuales ([www.louvre.fr](http://www.louvre.fr)), etc.

- **Aplicaciones web Transaccionales:** “Son creadas para proveer más interactividad, dándole al usuario la posibilidad no solo de interactuar con la aplicación de modo solo lectura, sino también realizar modificaciones en el contenido.” (Kappel, Pröll, Reich, & Retschitzegger, 2003). Son aplicaciones en las cuales los usuarios pueden realizar modificaciones en el contenido subyacente de la aplicación, desencadenar eventos y transacciones. Ejemplo muy conocidos son las tiendas electrónicas ([www.amazon.com](http://www.amazon.com), [www.vitalis.net/tiendavirtual.htm](http://www.vitalis.net/tiendavirtual.htm)), reservaciones de hoteles, compra de boletos aéreos, es decir todo lo relativo al comercio electrónico, banca electrónica, entre otros.
- **Aplicaciones Web basadas en el flujo de trabajo:** “Modelan y dar soporte a la manipulación del flujo de trabajo dentro o entre compañías, autoridades públicas y usuarios privados. La razón principal de estas aplicaciones es proveer servicios Web que garanticen la interoperabilidad de los entes. La complejidad de los servicios en cuestión es la autonomía de los empleados de la

compañía y la necesidad de contar con flujos de trabajo robustos y flexibles a la vez.” (Kappel, Pröll, Reich, & Retschitzegger, 2003), en otras palabras, son aplicaciones que proporcionan una herramienta para facilitar el flujo de trabajo de las compañías. Ejemplo de éstas aplicaciones son: business to business in e-commerce ([www.mercadolibre.com](http://www.mercadolibre.com), [www.ebay.com](http://www.ebay.com)), Aplicaciones del sector público (e-government: [www.seniat.gov.ve](http://www.seniat.gov.ve)), soporte para pacientes en aplicaciones del sector salud, sistemas de planificación y organización en línea, gestión de inventarios, etc.

- **Aplicaciones Web Colaborativas:** “Empleadas especialmente para la colaboración entre operaciones y entidades no estructuradas. Son necesarias para la comunicación entre los usuarios donde su colaboración es particularmente alta. La aplicaciones Web colaborativas apoyan los procesos para compartir información y espacios de trabajo donde se pueda administrar la información compartida.” (Kappel, Pröll, Reich, & Retschitzegger, 2003). En otras palabras, estas aplicaciones promueven la comunicación y alto nivel de interacción entre sus usuarios, así como herramientas donde se puedan crear, editar y la información compartida. Ejemplos de éstas aplicaciones son: Weblogs ([www.weblogs.com](http://www.weblogs.com), [www.weblogssl.com](http://www.weblogssl.com)),

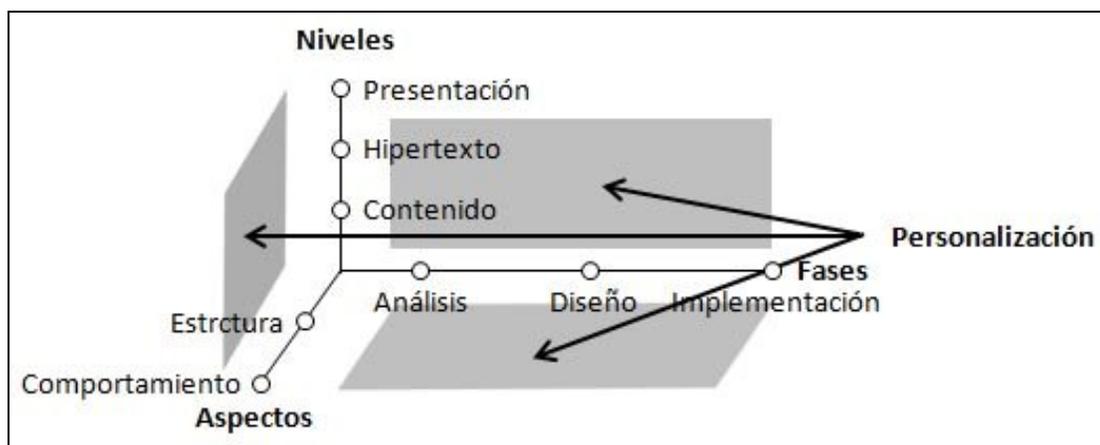
Chat rooms ([www.latinchat.com](http://www.latinchat.com)), plataformas e-learning ([www.aulaglobal.net.ve](http://www.aulaglobal.net.ve)).

- **Web sociales:** “Son aplicaciones donde los usuarios proveen su identidad (se registran) a una pequeña comunidad de otros usuarios con similares intereses.” (Kappel, Pröll, Reich, & Retschitzegger, 2003). Las Web sociales tienen las mismas características de las aplicaciones Web colaborativas, pero se diferencian en que las Web sociales persiguen un objetivo. Ejemplos de este tipo de aplicación son [www.facebook.com](http://www.facebook.com), [www.hi5.com](http://www.hi5.com), etc.
- **Portal Orientado a aplicaciones Web:** “Proveen un único punto de acceso a fuentes heterogéneas de información y servicios” (Kappel, Pröll, Reich, & Retschitzegger, 2003). Ejemplos de este tipo de aplicaciones son los portales que ofrecen motores de búsqueda ([www.google.com](http://www.google.com), [www.yahoo.com](http://www.yahoo.com)). Algunos portales nacieron como tiendas electrónicas, pero fueron evolucionando a los llamados *Marketplace portals* como es el caso de Amazon ([www.amazon.com](http://www.amazon.com)); están también los portales orientados a un tipo específico de usuarios (*community portals*).

## II.4 Marco de referencia para el estudio de las metodologías.

Para proceder a describir las metodologías es imprescindible explicar el marco de referencia que resumen las potencialidades y características de tales propuestas.

El marco de referencia es producto de revisiones y surveys identificados en la literatura, en especial (Fraternali, 1999) y (Retschitzegger & Schwinger, 2000) donde se identifican los siguientes elementos (ver figura N° 3).



**Figura N° 3.** Características a considerar en el desarrollo de aplicaciones Web.  
Tomado de (Kappel, Pröll, Reich, & Retschitzegger, 2003)

**Niveles:** “Representan los distintos puntos de vista desde los cuales se puede estudiar el sistema: contenido, navegación (hipertexto) y presentación” (Cachero, 2002)

Los niveles representan como está compuesta una aplicación Web, el contenido es la estructura lógica de la aplicación, el hipertexto es como está estructurado el contenido dentro de una página (nodo) y sus enlaces, y por último la presentación que es la interfaz del usuario, es decir, el diseño de la página. Para el caso de software tradicional, sólo aplicarían los niveles de: contenido y la presentación, ésta es la principal diferencia con las aplicaciones Web.

El diseño de cada uno de estos niveles de forma separada es lo que permite la reutilización, lo que ayuda a reducir la complejidad de este tipo de aplicación.

**Aspectos:** “añaden la perspectiva acerca del carácter estático (estructural) / dinámico (comportamiento) de cada uno de los niveles.” (Cachero, 2002). La relevancia que pueden tener los modelos de estructura y comportamiento dependen en gran medida del tipo de aplicación que se va a realizar, ya que si se desea desarrollar una aplicación estática, el modelado del comportamiento es irrelevante, en cambio, una aplicación que maneje diversos tipos de usuarios los cuales interactúan con el sistema, el modelado del comportamiento se hace necesario.

**Fases:** “niveles de abstracción que deberían tener soporte en cualquier propuesta de modelado para favorecer tanto la reflexión como el reuso”

(Cachero, 2002). En otras palabras, son las secuencias de los pasos para modelar los niveles, por ejemplo: la aplicación del análisis en modelar los niveles y los aspectos, luego aplicar el diseño en el modelado de los niveles, etc.

#### **II.4.1 Modelos fundamentales en el desarrollo de aplicaciones Web.**

Por las características propias de las aplicaciones Web y tomando en cuenta el marco de referencia anterior, es necesario que las propuestas metodológicas sean capaces de describir modelos específicos. Los modelos, producto de distintas actividades de las metodologías, deben ser más expresivos en el desarrollo Web, que aquellos generados para el desarrollo de aplicaciones no convencionales, vista la capacidad hipertextual de las primeras.

Los modelos principales que deben generarse en cualquier propuesta madura son:

##### **1. Modelado de Requerimientos:**

En esta etapa es donde se recogen, se evalúan y se analizan todas las características que se deben desarrollar para cumplir los objetivos que persigue la aplicación Web.

“Varias técnicas pueden ser usadas para identificar, analizar, describir, evaluar y gestionar los requerimientos de una

aplicación Web. Los casos de usos son la técnica de modelado más empleada en el levantamiento de los requerimientos, ya que puede ser representada gráficamente.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

A la hora de modelar una aplicación Web en donde no se tenga claro los objetivos que persigue, su alcance y las peticiones de los usuarios es importante contar con una metodología que proporcione las herramientas de modelado necesarias para levantar los requerimientos de la aplicación.

“Una peculiaridad de los requerimientos de una aplicación Web es la funcionalidad de la navegación, la cual permite al usuario desplazarse a través del hipertexto y encontrar nodos.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

Por tal motivo es importante que a la hora de levantar los requerimientos de una aplicación Web, se cuente con herramientas apropiadas para este tipo de aplicaciones y tomar herramientas de modelado de requerimientos genéricas para cualquier tipo de aplicación.

## **2. Modelado del Contenido:**

“La información provista por una aplicación Web es uno de los factores más importantes para el éxito de la aplicación, ya que originalmente la Web es un medio de información. Modelar contenido donde solo se incluya el modelado de la data normalmente es suficiente para aplicaciones Web estáticas. En aplicaciones Web complejas además del modelo de la data, se

requiere modelar los aspectos de su comportamiento. Esto significa que el modelado del contenido incluye la creación del modelo del dominio, consistente con los aspectos estáticos y dinámicos, conocido tradicionalmente como ingeniería de software.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

El modelado del contenido simplemente es la estructuración de cómo será almacenada la data de la aplicación (Base de Datos) y cómo será su comportamiento según la tarea que se ejecute en la aplicación.

Este tipo de modelado es fundamental a la hora de diseñar aplicaciones Web, debido a que un buen diseño del contenido facilitaría el proceso de desarrollo, así como la optimización de los recursos y haría más flexible a la aplicación a la hora de realizar modificaciones o nuevos requerimientos.

### **3. Modelado del Hipertexto:**

“La no linealidad del hipertexto es una de las propiedades más importantes cuando se está modelando una aplicación Web. Así la Estructura del hipertexto debe ser diseñada cuidadosamente. Esto puede ser resuelto utilizando estructuras de acceso adecuadas, Ejemplo: opciones de navegación, evitando así el riesgo de que los usuarios se pierdan y se expongan al excesivo estrés cognitivo” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

Al momento de diseñar una aplicación Web siempre es necesario modelar como los usuarios van a navegar a través de las páginas, este diseño de la navegación es lo que se define como el modelado del hipertexto. La realización de este modelado es fundamental a la hora de diseñar una aplicación Web, ya que es lo que la diferencia de las aplicaciones tradicionales.

“El objetivo del modelado del hipertexto, también conocido como modelo Navegacional, es especificar la navegación a través del contenido de una aplicación Web, Ejemplo: los links de navegación habilitados para los usuarios.

El modelado del hipertexto se enfoca en los aspectos estructurales del hipertexto y de los accesos a los elementos. El comportamiento de la navegación de una aplicación Web normalmente no es representado de manera explícita, porque provee muy poca información adicional a los desarrolladores.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

El modelado del hipertexto es prácticamente el mapa navegacional de la aplicación, así como la estructura del contenido dentro de los nodos. Un buen diseño del hipertexto permite que la aplicación sea más fácil de usar y evita la redundancia de la información, generando una aplicación más amigable.

#### **4. Modelado de Presentación:**

“Similar a la tradicional ingeniería de software. El modelo de presentación detalla cómo será la interfaz del usuario y el ‘look and feel’ de la aplicación Web.

El modelado de la presentación está destinado a diseñar la estructura y el comportamiento de la interfaz del usuario para asegurar que la interacción con la aplicación Web sea sencilla y se auto-explique. Adicionalmente, la comunicación y representación de tareas dentro de la aplicación Web son tomadas en cuenta.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

El modelado de la presentación de una aplicación Web, simplemente especifica y detalla cómo será la interfaz gráfica que interactuará con los usuarios. Este modelado genera 2 resultados, el primero es el de producir el modelado recurrente de la página (Ejemplo: el encabezado y el pie de página). El segundo es referente a la estructura de cada página de la aplicación, es decir, en donde irán colocados los botones, los cuadros de texto, la información, etc.

#### **5. Modelado de Personalización:**

“El modelo de personalización está dirigido explícitamente a la representación del contexto de la información y la adaptación derivada para eso.” (Kappel, Pröll, Reich, & Retschitzegger, 2003).

El modelo de personalización especifica cómo va a ser la estructura y el comportamiento de la aplicación según el usuario que interactúe en un momento específico con la aplicación.

Para este trabajo de grado se dividirá el modelado del contenido, hipertexto y presentación de la siguiente manera:

- Modelado de la estructura del contenido.
- Modelado del comportamiento del contenido.
- Modelado de la estructura del hipertexto.
- Modelado del comportamiento del hipertexto.
- Modelado de la estructura de la presentación.
- Modelado del comportamiento de la presentación.

## **II.5 Metodologías de desarrollo Web.**

Algunas metodologías de desarrollo de aplicaciones Web como OOHDM (Schwabe & Rossi, 2000), OO-H (Cachero & Gómez, 2003) y OOWS (Fons, Pastor, Valderas, & Ruiz, 2002) proponen todas las fases necesaria para el desarrollo de una aplicación Web, desde el levantamiento de los requerimientos hasta el desarrollo de la aplicación, no toman en cuenta la fase de prueba e implantación del aplicativo. UWE (Koch & Kraus, 2002), WebML (Ceri, Fraternali, & Bongio, 2000) y NDT (Escalona, Mejias, Torres, & Reina, 2002) solo contemplan las fases de levantamiento de

requerimiento, análisis y diseño. Algunas de éstas metodologías utilizan esquemas de modelado haciendo uso de notaciones estandares tales como UML y E-R. Otras proponen la combinación de notaciones propias y estandares.

A continuación se describen las metodologías de desarrollo Web que serán analizadas para la realización del modelo que propone este trabajo de grado.

### **II.5.1 HDM – Hypermedia Design Model**

“HDM prescribe la definición de un esquema de aplicación, que describe en general las clases de elementos de información en términos de sus características comunes de presentación, su estructura de organización interna y los tipos de sus interconexiones. Un esquema, por lo tanto, la captura semántica y estructural de las regularidades en la estructura de la presentación para una determinada clase de aplicaciones. Una vez que el esquema se ha especificado, el modelo también le permite definir una aplicación particular, mediante la aportación de primitivas para describir una instancia del esquema, es decir, casos reales de información de clases y tipos de conexión.” (Garzotto, Paolini, & Schwabe, 1993).

El modelo HDM (Garzotto, Paolini, & Schwabe, 1993) divide el proceso de diseño de una aplicación de hipertexto en 2 partes: el *authoring-in-the-large*, que se refiere a la especificación y diseño de los aspectos

globales y estructurales de la aplicación, y el *authoring-in-the-small*, que se refiere al desarrollo del contenido de los nodos, a más detalle.

El modelo HDM establece una serie de nociones (Primitivas) como: entidades, componentes, perspectivas, unidades, enlaces y esquema HDM.

**Entidades:** son cada una de las estructuras de la información dentro del nodo.

**Componentes:** Una entidad dentro de HDM (nodo) es una recopilación de componentes, es decir, que un componente es una abstracción de un conjunto de unidades, las cuales funcionan como contenedores de información.

**Perspectivas:** son las diferentes formas como puede ser mostrada una información dentro de la aplicación.

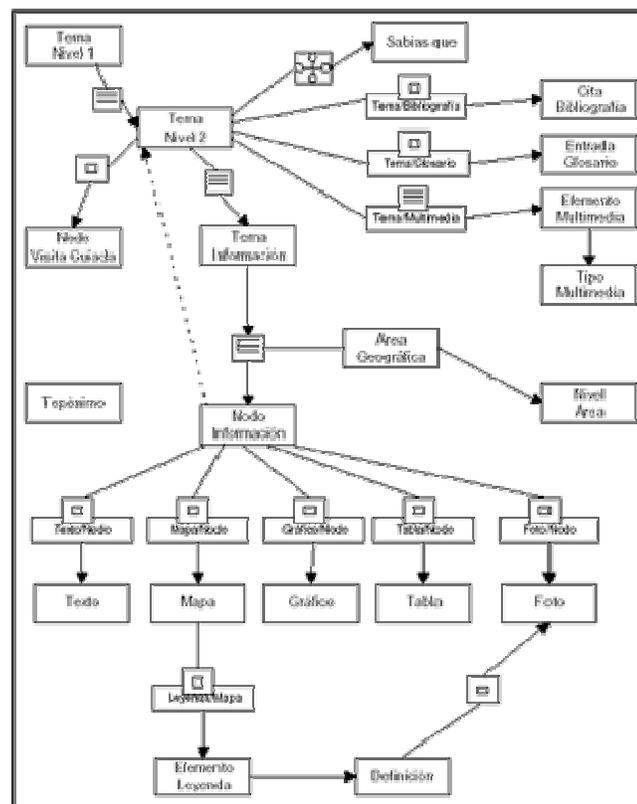
**Unidades:** Es un componente con una perspectiva determinada.

**Enlaces:** es la conexión entre los nodos, lo que genera el mapa navegacional de la aplicación. En HDM ese mapa es en forma de árbol de manera jerárquica, todo comienza desde un nodo raíz.

**Esquema HDM:**

Es el resultado del diseño de la aplicación en HDM, el esquema es definido así por sus autores (En la figura N°4 se muestra un ejemplo del esquema):

“Una especificación de HDM de una aplicación de hipertexto consiste en un esquema de definición y un conjunto de definiciones de los casos. Un esquema de definición especifica un conjunto de tipos de entidad y tipos de vínculos. Los casos son permitidos en una aplicación solo si ellos obedecen a las limitaciones especificadas por el esquema.” (Garzotto, Paolini, & Schwabe, 1993).



**Figura N° 4.** Esquema HDM. Tomado de Garzotto (Garzotto, Paolini, & Schwabe, 1993)

## II.5.2 RMM - Relationship Management Methodology

RMM (*Relationship Management Methodology*) (Isakowitz, Stohr, & Balasubramanian, 1995) es una metodología de desarrollo Web en el cual los elementos principales son el modelo E-R (Entidad-Interrelación) y el modelo RMDM (*Relationship Management Data Model*) basado en el modelo HDM.

RMM propone un lenguaje con el cual se describen los objetos del dominio de la aplicación, sus interrelaciones y los mecanismos de navegación dentro de la aplicación. RMM genera un esquema completo del dominio y de la navegación de la aplicación que se denomina esquema RMDM y se obtiene como resultado de las tres primeras etapas del método.

Las etapas son:

- **Primera etapa:** En esta primera etapa se representa la información del dominio, mediante un diagrama Entidad – Interrelación (E-R).
- **Segunda etapa:** En esta etapa, es donde se determina la manera de cómo estará estructurada la información dentro de las entidades, las cuales se presentaran al usuario y se determina la forma en la cual pueden acceder a ella, es decir, dividir una entidad en pedazos y organizarlos en una red hipertextual.

- **Tercera etapa:** En este paso, se diseñan los caminos que permitan la navegación del hipertexto. Cada una de las relaciones asociativas que aparecen en el diagrama de E-R se analiza

Las cuatro etapas restantes consisten en:

- definición del protocolo de conversión de elementos del diagrama RMDM en objetos de la plataforma de desarrollo
- concepción de la interfaz del usuario
- concepción del comportamiento en ejecución
- construcción del sistema y pruebas

Mediante el modelo de datos RMDM (*Relationship Management Data Model*), que se forma a partir de los diagramas E-R, se describirá no sólo la información referente a los objetos, sino también a la navegación entre ellos. En la figura N° 5 se muestra un ejemplo del modelo RMDM.

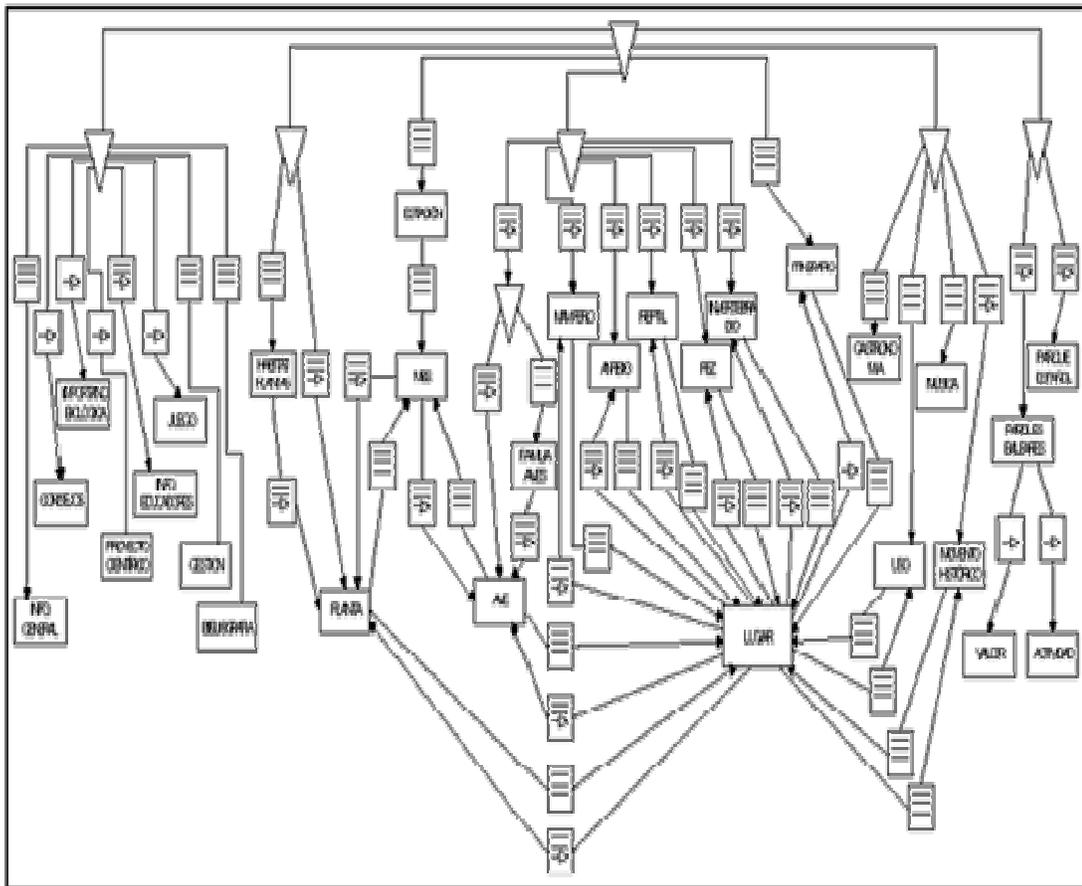


Figura N° 5. Diagrama RMDM. Tomado de (Isakowitz, Stohr, & Balasubramanian, 1995)

### II.5.3 OO-H – Object-Oriented Hypermedia

OO-H (Cachero, 2002) es una metodología que se basa en el paradigma orientado a objeto (OO), la cual ofrece a los desarrolladores las herramientas necesarias para la semántica y la notación de la aplicación Web.

OO-H toma el proceso de desarrollo de aplicaciones tradicionales, pero le agrega el complemento navegacional que es requerido en todas las

aplicaciones Web, este complemento es: el diagrama de acceso navegacional (DAN).

**El Diagrama de Acceso Navegacional:** Para obtener el modelo navegacional de la aplicación, OO-H propone la realización de uno o varios DAN con diferentes enfoques de las vistas de la aplicación, esto dependerán de los tipos de usuarios que interactuaran con el sistema. El diagrama se divide en 4 tipos de constructores: las clases de navegación, los objetivos de navegación, los vínculos de navegación y las colecciones.

#### **Actividades (fases) realizadas por OO-H:**

1. **Modelo Requisitos Funcionales:** para esta actividad la metodología propone una técnica de ejecución dentro del campo de la ingeniería de requisitos, el cual proporciona como resultado de esta actividad un diagrama de casos de usos UML, que constituye el modelo de requisitos funcionales.
2. **Modelo Análisis Navegación:** Una vez que se tenga el diagrama de uso de la fase anterior, el análisis de la navegación agrega un conjunto aspectos que dan origen al diagrama de navegación semántica (DNS). En esta fase se captura la estructura de alto nivel de la navegación del usuario en la aplicación, donde se debe tener en cuenta el posible propósito de dicho usuario a la hora de diseñar los

menús de acceso, para lograr así diseñar un modelo de hipertexto eficaz.

3. **Modelo Dominio:** En esta fase se genera un diagrama de clases (DC) UML, en donde se incluyen las clases, atributos, métodos y las relaciones entre ellas. En una primera etapa se genera un DC general, y luego se va refinando este diagrama para recoger detalles de más bajo nivel, como es el tipo de los atributos, parámetros de entrada y salida de los métodos, precondiciones y pos condiciones de invocación, entre otros.
4. **Modelo Diseño Navegación:** Es en esta fase donde se construye un modelo de diseño de la navegación, es donde va a estar representado el contenido, la navegación y la funcionalidad que componen la interfaz de la aplicación. En este modelo es donde se captura todo el contenido y la navegación de la aplicación necesaria para la generación del primer prototipo. Además se recoge la información sobre los caminos navegacionales, restricciones de navegación e interfaces de servicio, que tienen como fin dar respuesta a cada requisito funcional de la aplicación.
5. **Modelo Diseño Presentación:** En esta última fase se generan dos modelos: Diagrama de Presentación Abstracta (DPA), el cual muestra una visión de la estructura lógica de la aplicación, y permite modificar

la estructura con elementos como: el pie de página y el encabezado. El segundo modelo es el Diagrama de Diseño Visual (DDV), el cual permite editar cada una de las páginas que conforman la estructura para especificar aspectos como colores, posicionamiento o tipos de constructor lógico.

#### **II.5.4 EORM – Enhanced Object Relationship Methodology**

EORM (Lange, 1996) es una metodología orientada a objeto (OO) con un modelado de proceso iterativo.

Esta metodología consta de las siguientes fases:

- 1. Fase de Análisis:** en esta fase se genera un modelo de clases de objetos con atributos, los métodos y las relaciones entre las clases. En esta fase se trata de orientar a objeto la aplicación sin tomar en cuenta los aspectos hipermediales.
- 2. Fase de Diseño:** En esta fase se modifica el modelo obtenido en la fase anterior, al cual se le agrega la semántica apropiada a las relaciones entre las clases de objetos, los cuales son convertidos en enlaces hipermediales. El resultado de esta fase son los esquemas EORM, en donde se refleja la estructura de la información y las posibilidades de navegación que ofrecerá la aplicación.

3. **Fase de Desarrollo:** En esta fase es donde se lleva a cabo la codificación de la aplicación, es decir, se transforman los esquemas en código, se genera la base de datos y se elaboran los formularios de consultas de las clases identificadas.

### **II.5.5 OOHDM - Object-Oriented Hypermedia Design Method**

OOHDM (*Object-Oriented Hypermedia Design Method*) (Schwabe & Rossi, 1995) es una metodología que se basa en el paradigma orientado a objeto (OO) y permite una descripción precisa de los elementos de información compleja que manejará la aplicación a ser diseñada. También se pueden especificar los patrones complejos de navegación y las transformaciones de la interfaz del usuario.

OOHDM propone 4 etapas (fases) y consideran el paradigma de proceso incremental. Cada una de las etapas se enfoca en un diseño particular de una preocupación, en donde el resultado será un modelo OO. OOHDM es una extensión de HDM, pero lo que la distingue claramente es el proceso de concepción orientado a objetos.

A continuación se enumeran las 4 etapas propuestas por OOHDM:

1. **Diseño Conceptual:** en esta etapa se construye un esquema conceptual que representa los objetos (clases), sus relaciones y colaboraciones existentes entre ellos. Las clases se describen en modelos OO con sus respectivos atributos.
2. **Diseño Navegacional:** esta etapa es considerada como crítica dentro de OOHDM, ya que la navegación es un paso fundamental en el diseño de una aplicación Web. En esta etapa se genera un modelo de navegación el cual está construido por “vistas” sobre el modelo conceptual, lo que permite la elaboración de diferentes modelos de acuerdo a los distintos usuarios que interactuarán con la aplicación. Cada modelo de navegación ofrece una vista subjetiva del modelo conceptual.

El modelo de navegación consta de 2 esquemas, el esquema de clase de navegación y el esquema contextual de navegación. OOHDM hereda de HDM el conjunto de tipos de clases de navegación pre-definidos: nodos, enlaces y la estructura de acceso, así como la semántica de los nodos y enlaces.

3. **Diseño de interfaces abstractas:** en esta etapa es donde se especifica la interfaz abstracta de la aplicación, es decir, definir la forma en la cual los objetos de navegación serán mostrados, así como

qué interfaz del objeto de navegación será activado y que transformación de la interfaz tendrá lugar y en qué momento.

4. **Implementación:** esta etapa es donde en realidad se aplicará el diseño, particularmente se considera el tipo de la ejecución de la aplicación. OOHDMM se enfoca en como los diseños pueden ser aplicados en la WWW.

## **II.5.6 SOHDM – Scenario-based Object-Oriented Hypermedia Design**

### **Methodology**

SOHDM (Lee, Lee, & Yoo, 1998) es una metodología que sigue el paradigma OO y que se basa en escenarios, los cuales se utilizan para mejorar la capacidad expresiva de modelado. Las vistas OO son utilizadas como unidades de navegación, así como vistas lógicas del usuario.

SOHDM propone las siguientes 6 fases:

1. **Análisis del dominio:** se indican los límites de la aplicación que se va a desarrollar y se representan mediante un diagrama de contexto, que no es más que un diagrama de flujo de datos (DFD). En esta fase se identifican los eventos de las entidades externas que interactuarán con la aplicación, por ejemplo: una acción la cual inicia la ejecución del sistema. SOHDM propone la utilización de escenarios para

identificar los requerimientos de la aplicación, estos escenarios son representados mediante los *Scenarios activity charts* (SACs). Los SAC describen los procesos del negocio según los tipos de usuarios que interactuarán con la aplicación.

2. **Modelo de Objetos:** Los SACs generados en la fase anterior son usados para modelar objetos. Estos escenarios son transformados en objetos en forma de las tarjetas CRC (Clase-Responsabilidad-Colaboración). Los usuarios son los principales candidatos a ser objetos, donde también se toman en cuenta las actividades. Luego que los objetos son identificados estos son expresados en un documento en el cual se incluyen los atributos y asociaciones de cada uno así como la cardinalidad de la asociación.
3. **Diseño de la vista:** En esta fase los objetos son organizados en unidades de navegación, en donde cada unidad representa una vista OO. La utilización de vistas en el diseño de una aplicación Web tiene varias ventajas: la primera es que las vistas pueden soportar varios usuarios los cuales tienen diferentes requerimientos, la segunda es que las vistas pueden reducir las características cognitivas de una aplicación Web, ya que las diferentes responsabilidades y atributos de los objetos se pueden agrupar dentro de las vistas.
4. **Diseño Navegacional:** en esta fase se diseña la forma cómo los usuarios utilizan y aglutinan la información sobre la base de los

escenarios. En una aplicación Web, la manera como los usuarios exploran la hipermedia es el factor más relevante dentro del proceso de diseño, ya que un buen diseño de navegación evitaría la información redundante y ayuda a prevenir que el usuario se pierda en el hiperespacio. En esta fase las vistas OO y los nodos de la estructura de acceso (ASN) son adoptados por las unidades de navegación. Los ASN se utilizan para agrupar las características y funciones de los diferentes actores de la aplicación, lo cual permite a los usuarios acceder a otras partes de la aplicación. También proporcionan a los usuarios el acceso a las estructuras que pueden utilizar.

5. **Diseño de Implementación:** en esta fase se genera la estructura de la página, el flujo de la página, la interfaz de usuario, la lógica y esquema de base de datos para la construcción de un entorno de desarrollo.
6. **Construcción:** En este paso los desarrolladores generan una aplicación hipermedia, la cual cumple con todas las características especificadas en las fases anteriores.

## II.5.7 WebML – Web Modeling Language

WebML (*Web Modeling Language*) (Ceri, Fraternali, & Bongio, 2000) es una metodología que consta de 4 etapas soportadas bajo un modelado de proceso iterativo, que le permite a los diseñados expresar características fundamentales de una aplicación Web. WebML está compuesta por modelos con una intuitiva representación gráfica para ayudar a los miembros del equipo de desarrollo que no manejan los términos técnicos.

WebML propone realizar la fase de análisis y diseño en cuatro (4) etapas:

1. **Modelo Estructural:** en esta fase se expresa el contenido que tendrá la aplicación Web, en términos de entidades e interrelaciones E-R. WebML no propone un lenguaje de modelado para la data, pero es compatible con las clásicas notaciones como el modelo E-R y el diagrama de clases de UML.
2. **Modelo de Hipertexto:** En esta etapa se describe qué páginas y contenidos componen la aplicación Web y cómo están enlazadas a través de dos sub-modelos:
  - **Modelo de Composición:** es en donde se especifican que páginas componen el hipertexto y cada unidad de contenido que componen las páginas. WebML propone 6 tipos de unidades de contenido: datos, multi-datos, índice, filtro, desplazador y directa

unidades. Los datos se utilizan para publicar la información de un sólo objeto, mientras que los restantes tipos de unidades representan formas alternativas para navegar por un conjunto de objetos

- **Modelo de Navegación:** es donde se expresa la manera en la cual las páginas están vinculadas con el contenido de las unidades, dando forma al hipertexto.

3. **Modelo de Presentación:** en esta etapa se define la diagramación y la apariencia gráfica de las páginas. Las especificaciones de la presentación son de una página en específico o en general.

4. **Modelo de Personalización:** los tipos de usuarios que interactuarán con la aplicación son modelados explícitamente en un esquema estructural, donde cada entidad del esquema pertenece a cada tipo de usuario identificado. Las características de éstas entidades se pueden utilizar para almacenar un grupo específico de contenido

### **II.5.8 W2000**

W2000 (Baresi, Garzotto & Maritati, 2002) es una metodología que está organizada en paquetes que se van desarrollando en un proceso de 4 fases que componen los modelos necesarios para el diseño de una

aplicación Web. W2000 toma como notación los diagramas propuestos por UML.

A continuación se muestran las 4 fases propuestas por W2000:

1. **Análisis de requerimientos:** en esta fase se identifican los límites, objetivos y requerimientos funcionales de los procesos del negocio para el cual va dirigida la aplicación Web. Los requerimientos funcionales se levantan según la percepción de las diferentes clases de usuarios.
2. **Diseño de hipermedia:** en esta fase se realiza el diseño de la información, la navegación y la presentación. Comienza con una redacción de cada uno de estos aspectos y posteriormente se generan modelos con la estructura de cada uno de ellos. Estos modelos se van refinando para introducir todos los detalles que se deben establecer antes de implementar la aplicación.
3. **Diseño de los servicios:** esta fase se ejecuta en paralelo con la fase de diseño del hiperespacio y especifica las principales transacciones del negocio que va a soportar la aplicación a desarrollar. Dicha especificación está orientada al usuario
4. **Actividades de personalización:** se realizan en caso de ser necesario. Se definen las características que se van a personalizar y

posteriormente se introducen en los modelos generados en las fases anteriores.

### **II.5.9 UWE – UML-based Web Engineering**

UWE (Koch & Kraus, 2002) es una metodología con un modelado de proceso espiral / interactivo – incremental y tomando como lenguaje de notación UML, ésta metodología está enfocada en el diseño sistemático, la personalización y la generación semiautomática de escenarios. UWE utiliza vistas especiales soportadas por los diagramas gráficos de UML, como el modelo de navegación y el modelo de presentación. UWE no limita el número de vistas de una aplicación, ya que los diseñadores también pueden hacer uso de otra técnica de modelado UML para agregar otras vistas a la aplicación.

UWE propone los siguientes pasos para el desarrollo de una aplicación Web:

1. **Especificación de requerimientos:** es donde se describen los requisitos funcionales de la aplicación a desarrollar. UWE propone el modelo de casos de uso de UML para el levantamiento de los requerimientos, ya que a través de esta herramienta se puede describir una parte del comportamiento de la aplicación sin revelar la

estructura interna, así como la identificación de los distintos usuarios que interactuarán con la aplicación.

2. **Modelo Lógico-Conceptual:** en este paso se especifican los elementos del dominio de la aplicación. UWE propone para este paso la utilización de un diagrama de clases de UML
3. **Modelo de Navegación:** en este paso se genera la especificación de los objetos que pueden ser visitados mediante la navegación dentro de la aplicación Web y las asociaciones entre ellos. En UWE los modelos de la navegación son representados por los diagramas de clases estereotipadas de UML.
4. **Modelo de presentación:** en este paso se define como los usuarios visualizarán los objetos de navegación y las primitivas de acceso. Para este modelo UWE propone una forma particular de un diagrama de clase, el cual representa de forma gráfica, las clases definidas anteriormente.

#### **II.5.10 WSDM – Web Services Distributed Management**

WSDM (De Troyer & Leune, 1998) es una metodología para el diseño de aplicaciones Web centrado en los usuarios, por lo que el primer paso en esta metodología es la identificación de los tipos de usuarios que interactuarán con la aplicación que se va a diseñar. Una vez identificados los

usuarios se procede a describir sus características y sus requerimientos, lo que WSDM denomina perspectivas. Las perspectivas son la entrada de las actividades que se realizarán en la fase de diseño.

WSDM consta de las siguientes fases:

1. **Modelo de usuario:** se identifican tipos de usuarios que interactúan en la aplicación, esta fase se divide en 2 etapas: la clasificación de usuarios, en donde se identifican y clasifican los posibles usuarios de la aplicación y la descripción de los grupos de usuarios, es donde se analizan los tipos de usuarios identificados en la fase anterior, especificando sus necesidades de información y sus características generales.
2. **Diseño conceptual:** esta fase se divide en 2 etapas: el modelado de objetos, en donde se identifican las necesidades de información de las distintas clases de usuarios y en donde sus perspectivas serán descritas formalmente. La segunda etapa es el diseño de la navegación, donde se define el mapa navegacional de la aplicación.
3. **Diseño de implementación:** en esta fase se define el “look and feel” de la aplicación Web, es decir, la interfaz del usuario.
4. **Implementación:** en esta fase se lleva a cabo la codificación de la aplicación diseñada.

### **II.5.11 OOWS – Object Oriented Web Solutions**

OOWS (Fons, Pastor, Valderas, & Ruiz, 2002) es una propuesta metodológica de desarrollo de aplicación Web, basada en el paradigma orientado a objeto (OO). OOWS toma como base OO-Method (Pastor, Insfrán, & Pelechano, 1999) y agrega modelos para definir las características navegacionales y de presentación de la aplicación.

OOWS propone un proceso de desarrollo de aplicaciones web con dos fases principales:

1. **Especificación del problema:** en esta fase se capturan los requerimientos funcionales del sistema mediante el diagrama de caso de usos (UML) y posteriormente el modelo conceptual de la aplicación (UML). En esta fase se generan los siguientes modelos:
  - Modelo de Objetos: define la estructura y las relaciones entre las clases identificadas.
  - Modelo Dinámico: se describen las posibles secuencias de los servicios.
  - Modelo Funcional: capta la semántica relacionada a los cambios de estado entre los objetos.
  - Modelo Navegacional: define el mapa de navegación asociado a las clases de los objetos del modelo. En etapa se define la navegación permitida por los usuarios de la aplicación.

- **Modelo de Presentación:** Se define de manera abstracta la estructura lógica de presentación de los objetos navegacionales en la interfaz de usuario.

En esta fase también se identifican los tipos de usuarios que interactuarán con la aplicación, y se especifica una presentación y navegación para cada uno de ellos.

2. **Desarrollo de la aplicación:** En esta fase se lleva a cabo el desarrollo del código de la aplicación que se diseñó en la fase anterior.

#### **II.5.12 NDT – Navigational Development Techniques**

NDT (Escalona, Mejias, Torres, & Reina, 2002) es una propuesta metodológica que se centra en las primeras fases del ciclo de vida y ofrece un proceso para especificar, analizar y diseñar los aspectos de navegación de la aplicación.

NDT propone un proceso de desarrollo dividido en dos fases:

1. **Captura y definición de requisitos:** en esta fase se especifican los requerimientos funcionales de la aplicación, para ello, NDT propone empezar estudiando el entorno de trabajo y definiendo los objetivos. Posteriormente se obtienen

las necesidades de almacenamiento de la información, es en esta etapa donde se define la estructura que debe tener la información que se almacena. En la tercera etapa de esta fase se realiza un estudio para identificar los tipos de usuarios que interactuarán con la aplicación, clasificándolos, definiendo sus roles posibles, sus incompatibilidades y relaciones de herencia.

La cuarta actividad que se propone en esta fase es el estudio de las necesidades funcionales del sistema, ya que se debe indicar lo que ofrecerá la aplicación a cada uno de los diferentes roles definidos en la etapa anterior. En la quinta y última actividad de esta fase se deben definir los requisitos de interacción, es decir, cómo el usuario va a interactuar con la aplicación.

2. **Análisis:** es en esta fase donde se establece el diseño del mapa navegacional de la aplicación, para cada tipo de usuario identificado. Para el diseño de la navegación NDT propone la aplicación del modelo conceptual de UML.

Por último, para concluir la breve explicación realizada a cada una de las propuestas metodológicas tomadas en cuenta para la realización del

modelo propuesto en este trabajo de grado, se presenta un resumen con las principales características de cada una de las propuestas (Tabla 1).

Metodología	Proceso del Desarrollo de la Aplicación	Técnica de Modelado	Modelos gráficos	Notación
<b>HDM</b>	1.-authoring-in-the-large 2.-authoring-in-the-small	Entidad - interrelación	1.-Diagrama Entidad - Interrelación	1.-Entidad - Interrelación
<b>RMM</b>	1.-Diseño Entidad-Relación 2.-Diseño Estructural de las Páginas de la Aplicación 3.-Diseño de Navegación 4.-Diseño de la Interfaz de Usuario 5.-Diseño del Comportamiento en Tiempo de Ejecución 6.-Construcción y Prueba de la Aplicación	Entidad - interrelación	1.-Diagrama Entidad - Interrelación 2.-Diagrama de Estructuración de las Páginas 3.-Diagrama de Datos de Administración de Relaciones	1.-Entidad - Interrelación 2.-Notificación Propia del Modelo
<b>EORM</b>	1.-Clases de Desarrollo de la Aplicación 2.-Composición del Entorno del Desarrollo de la Aplicación 3.-Entorno de Desarrollo de la Interfaz de Usuario	Orientado - Objeto	1.-Diagrama de Clases 2.-Diseño de la Interfaz del Usuario	1.-Técnica de Modelado de Objetos
<b>OOHDM</b>	1.-Diseño Conceptual 2.-Diseño Navegacional 3.-Diseño abstracto de la Interfaz de Usuario 4.-Implementación	Orientado - Objeto	1.-Diagrama de Clases 2.-Diagrama Navegacional, Clases - Contexto 3.-Diagrama de Configuración de la Vista de Datos Abstracta 4.-Diagrama de la Vista de Datos Abstracta	1.-Técnica de Modelado de Objetos 2.-UML 3.-Vista de Datos Abstracta 4.-Notificación Propia
<b>SOHDM</b>	1.-Análisis del Dominio 2.-Modelo en OO (Orientado Objeto) 3.-Diseño de la Vista 4.-Diseño Navegacional 5.-Diseño Implementación 6.-Construcción	Escenarios Vistas - OO	1.-Diagramas de Escenarios de Actividad 2.-Diagrama de Estructura de Clase 3.-Vista OO 4.-Esquema de Enlace Navegacional 5.-Esquema de Páginas	<b>1.-Notificación Propia</b>
<b>UWE</b>	1.-Especificación de Requerimientos 2.-Diseño Lógico Conceptual 3.-Diseño Navegacional 4.-Diseño de Interacción 5.-Diseño de Escenarios Web	Orientado - Objeto	1.- Diagramas UML	<b>1.-UML</b>
<b>WSDM</b>	1.-Modelado del Usuario 2.-Diseño Conceptual 2.1.-Modelo Objeto 2.2.-Diseño Navegacional 3.-Diseño Implementación 4.-Implementación	Entidad - interrelación Orientado - Objeto	1.-Diagrama Entidad - Interrelación o Clases 2.-Capas de Navegación	1.-Entidad - Interrelación / Técnica de Modelado de Objetos 2.-Notificación Propia
<b>OOWS</b>	1.-Especificación del Problema 2.-Desarrollo de la Aplicación	Orientado - Objeto	1.-Modelo de Objetos 2.-Modelo Dinámico 3.-Modelo Funcional 4.-Modelo de Navegación 5.-Modelo de Presentación	1.-UML 2.-Notificación Propia
<b>NDT</b>	1.-Captura y definición de requisitos 2.-Análisis	Orientado - Objeto	1.-Patrones 2.-Diagrama Caso de Usos 3.-Diagrama de Clases 4.-Diagramas de contextos Navegacionales	1.-UML 2.-Notificación Propia

**Tabla 1.** Comparativa de los aspectos relevantes de las metodologías de desarrollo Web.

## II.6 Herramientas CASE

Debido a la importancia que han alcanzado este tipo de instrumentos en el desarrollo de software, se decidió tomar en cuenta las características principales que poseen las herramientas CASE propuestas por las metodologías estudiadas en este trabajo de grado. Las herramientas CASE tendrán un papel fundamental a la hora de seleccionar la mejor metodología de desarrollo para una aplicación Web.

Las herramientas CASE son instrumentos de Ingeniería de Software Asistida por Computadora, es decir, son aplicaciones que tienen como objetivo el aumento de la productividad del equipo de trabajo, en el proceso de desarrollo de software, ayudando a reducir los tiempos de ejecución y costos del proyecto.

Los beneficios más importantes por lo cual un equipo de desarrollo debería contar con una herramienta CASE, según (Kendall & Kendall, 2005) y (Pressman, 2005), son las siguientes:

1. La integración de las actividades que debe realizar el equipo de desarrollo con las fases del ciclo de vida propuesto por la metodología que están utilizando.

2. La disminución del esfuerzo y del tiempo de ejecución para efectuar actividades globales, como: el control de calidad y la producción de documentos
3. El aumento del control de la ejecución del proyecto, ya que se puede lograr una mejor planificación y un monitoreo más preciso de las actividades realizadas.
4. La mejora de la comunicación entre los miembros del equipo de trabajo que está llevando a cabo el proyecto.

Las herramientas CASE deberían ofrecer ciertas funcionalidades, según (Kendall & Kendall, 2005) y (Pressman, 2005), para que puedan cumplir con el objetivo de el aumento de la productividad y la disminución de tiempo de ejecución y costo del proyecto. A continuación se mencionan las principales funcionalidades:

- Proporcionar un mecanismo para compartir la información guardada y generada de un proyecto.
- Poseer mecanismos que en caso de producirse un cambio en un modelo, este cambio se replique en los modelos que se generan a raíz del cambio.
- Proporcionar un control de versiones.
- Establecer un apoyo automatizado para el modelo de proceso de la metodología que se está usando.

- Dar soporte a la comunicación entre los miembros del equipo de trabajo.
- Proporcionar mecanismos para la verificación de la coherencia del diseño de la aplicación.
- Poseer mecanismos que generen parte del código de la aplicación y de la interfaz gráfica.

Por lo expuesto anteriormente, las características que serán tomadas en cuenta en este trabajo de grado para analizar las diferentes herramientas CASE de las metodologías de desarrollo Web son las siguientes:

- Integrado con las actividades de las metodologías de diseño.
- Contener controles de coherencia del diseño de la aplicación.
- Manejo las versiones de los diseños realizados.
- Auto generación de parte de la interfaz y código de la aplicación.
- Auto generación de parte de la documentación de la aplicación.

A continuación se muestra el cuadro comparativo (Tabla 2) de las herramientas CASE, que proporciona cada metodología de desarrollo Web estudiadas en este trabajo de grado, con respecto a las características nombradas anteriormente.

<b>CASE TOOL</b>	<b>Integrado con las actividades de las metodologías de diseño</b>	<b>Controles de coherencia del diseño de la aplicación</b>	<b>Manejo las versiones de los diseños realizados</b>	<b>Auto generación de parte de la interfaz y código de la aplicación</b>	<b>Auto generación de parte de la documentación de la aplicación</b>
OO-H (CAWE o VisualWADE )	Si	Si	Si	Si	Si
OOHDM (OOHDM-Web)	Si	Si	Si	Si	Si
WebMI (WebRatio)	Si	Si	No	Si	Si
OOWS (OOWS Suite)	Si	Si	No	Si	Si
UWE (ArgoUWE)	Si	Si	No	Si (Semi)	Si (Semi)
NDT (NDT-Tool)	Si	Si	Si	Si (Interfaz)	Si
WSDM (no cuenta con una herramienta CASE)	EP	EP	EP	EP	EP

**EP = En proceso. Información no disponible**

**Tabla 2.** Comparaciones de las herramientas CASE de las diferentes metodologías de desarrollo Web estudiadas

## **II.7 Criterios relacionados a los equipo de trabajo que serán usados en el modelo**

A continuación se mencionan y explican las características del equipo de trabajo que se deben tomar en cuenta a la hora de seleccionar la metodología de desarrollo para la realización de una aplicación Web.

### **II.7.1 Experiencia en desarrollo Web**

La primera características que se debe tomar en cuenta es si el equipo que estará a cargo del desarrollo Web tiene experiencia en el diseño de este tipo de aplicaciones, esto debido a que las aplicaciones Web tienen características especiales, que van desde el entorno en el que operan hasta los requerimientos de usuario. (Koch & Kraus, 2002).

### **II.7.2 Experiencia en modelos de procesos**

La segunda característica fundamental que debe conocer el equipo de trabajo es la experiencia que poseen en los modelos de procesos que son según (Sommerville, 2002) y (Pressman, 2005), la representación reducida del proceso que se llevará a cabo para la realización de una aplicación, es decir, es una abstracción del proceso real de realización del software.

Los modelos de procesos tiene como objetivo principal el de resolver los problemas que se presentan en las organizaciones. Por otra parte, los

equipos de desarrollo deben tener estrategias que los guíen en el proceso de construcción, esa estrategia es la que recibe el nombre técnico de modelo de proceso.

En la Tabla 3 se muestran los modelos de procesos más usados, con una breve explicación.

<b>Modelo de Proceso</b>	<b>Características Principales</b>
El modelo lineal secuencial (cascada)	El modelo posee un enfoque secuencial y sistemático para la construcción de aplicaciones, el cual comienza en un punto de la aplicación y va progresando con el análisis, diseño, codificación, pruebas e implantación.
El modelo de construcción de prototipos	El modelo se basa de en la recolección de los requerimientos del cliente, luego se genera un prototipo rápido y se le muestra al usuario, se recogen los comentarios y se inicia otra vez el proceso. El proceso se realiza hasta la aprobación del prototipo final.
El modelo DRA	El modelo es una adaptación del modelo lineal secuencial, orientado al desarrollo rápido de una aplicación mediante la utilización de componentes.
El modelo incremental	El modelo combina los elementos del modelo lineal secuencial, con la filosofía interactiva de construcción de prototipos.
El modelo espiral	El modelo posee un proceso de software evolutivo que mezcla la naturaleza iterativa de construcción de prototipos con los aspectos sistemáticos del modelo lineal secuencial.
El modelo de desarrollo concurrente	El modelo propone un proceso en forma de esquema como una serie de actividades técnicas importantes, tareas y estados asociados a ellas.
El modelo de desarrollo basado en componentes.	El modelo incorpora características del modelo en espiral. Es evolutivo por naturaleza y tiene un enfoque iterativo para la creación del software. El modelo está basado en el desarrollo por componentes.
El modelo de métodos formales.	El modelo comprende una serie de actividades que conducen a la especificación matemática de la aplicación.

**Tabla 3.** Modelos de procesos

### II.7.3 Experiencia en paradigma de modelado

La tercera característica que debe conocerse acerca del equipo de desarrollo de una aplicación Web, es en cuáles de los paradigmas de modelado tienen experiencia. Las metodologías de desarrollo Web siguen paradigmas de modelados para representar el diseño conceptual (contenidos, navegación, lógica del negocio, presentación, etc.) de una aplicación, Los paradigmas “son utilizados en una variedad de escenarios, como la documentación de las necesidades de los negocios y especificación de los aspectos de una aplicación” (Markovic & Pereira, 2007), entre los paradigmas de modelado más comunes en las propuestas metodológicas de desarrollo Web se encuentra:

1. DB (Orientado a Datos) Una de las propuestas metodológicas estudiadas que integra este paradigma es WebML.
2. O-O (Orientado a Objetos). Metodologías propuestas como: OO-H, OOHD, UWE, OOWS, entre otras; se basan en el paradigma OO.
3. H-T (Orientado a Hipertexto). WSDM es una propuesta metodológica que está realizaba bajo este paradigma

## **II.7.4 Experiencia en lenguaje de modelado**

Por último, los lenguajes de modelado son “la notación (principalmente gráfica) de que se valen los métodos para expresar los diseños” (Booch, Rumbaugh y Jacobson, 2006), es decir, que son un conjunto de símbolos y reglas que se utilizan para definir el diseño de una aplicación. Son muy útiles para mejorar la comunicación entre los interesados y los desarrolladores. Por lo tanto, poseer experiencia en estos lenguajes pudiera mejorar los tiempos y la calidad del diseño de la aplicación. A continuación mencionamos los lenguajes de modelados más usados en las metodologías de desarrollo de software:

1. Entidad Interrelación (E-R).
2. UML (Booch, Rumbaugh, & Jacobson, 2006).
3. Lenguajes Z.
4. Lenguajes propios de las metodologías, que a veces son derivaciones de los otros lenguajes de modelado.

### **CAPÍTULO III: Marco Metodológico**

En este capítulo se detallan cada uno de los aspectos relacionados con la metodología utilizada para desarrollar este trabajo, en otras palabras, evidenciar el “cómo” se llevó a cabo la investigación. Las fases que se siguieron fueron:

1. Revisión bibliográfica: se realizó la investigación de los antecedentes relacionados con el planteamiento de este trabajo de grado. La idea inicial fue el problema que tienen los desarrolladores de aplicaciones Web durante el diseño, ya que un gran porcentaje de este tipo de aplicaciones son desarrolladas haciendo uso de métodos “ad-hoc” o métodos convencionales de la ingeniería del software, modificados y ajustados, pero no alineados a la ingeniería Web. A partir de esta idea se empezó un proceso de revisión bibliográfica, así como una serie de entrevistas a profesionales con amplia experiencia en el desarrollo de aplicaciones Web.

Producto de esta fase, se generaron las siguientes conclusiones con respecto al por qué la gran mayoría de los desarrolladores producen aplicaciones Web de mediana calidad:

- a. Los equipos de trabajo desconocen las diferencias entre el desarrollo de software convencional asistido por los procesos de la ingeniería del software Vs los desarrollos Web, asistidos por la ingeniería Web.
- b. La mayoría de los equipos de trabajo tienen poca experiencia en el modelado de la navegación en el desarrollo de aplicaciones Web.
- c. La mayoría de los equipos de trabajo no conocen de la existencia de metodologías de desarrollo Web, o desconocen los aspectos relevantes de las metodologías.

Por último en esta fase se realizó el planteamiento del problema que se pretende minimizar, y se definió el objetivo general de este trabajo de grado.

2. Identificación del contexto y requerimientos para la definición del modelo: en esta fase se especificaron los pasos a realizar para la obtención de la información necesaria y qué aspectos considerar para realizar el modelo propuesto por este trabajo.

A continuación las etapas llevadas a cabo en esta fase. En la Tabla 4 se muestran los resultados obtenidos en cada una de las etapas:

1. Determinar los conceptos básicos del ambiente Web.
2. Especificar el alcance de este trabajo de grado, precisando cuales aspectos no van a ser tomados en cuenta.

3. Recopilar la información necesaria que servirá como base para la realización del trabajo de grado.
4. Especificar la información relevante de las metodologías de desarrollo Web y los criterios relacionados a los equipos de trabajo a tomar en cuenta.

Etapa	Actividades / Resultados
1	a. Se aclararon los términos que son usados con frecuencia en las aplicaciones Web. Algunos de estos conceptos son los siguientes: hipertexto, hipermedia, hiperdocumento, contenido, presentación, navegación, entre otros.
2	a. Se especificaron los objetivos que abarca esta investigación. b. Se explicaron brevemente los modelos de referencia de Hipermedia, los modelos de hipermedia, para establecer la diferencia entre ellos, y las metodologías de desarrollo de aplicaciones Web.
3	a. Se determinaron las dimensiones de modelado que requiere una aplicación Web como un paso dentro del proceso de desarrollo. b. Se determinaron y explicaron las características de los diferentes tipos de aplicaciones Web que existen actualmente.
4	a. Se identificaron las metodologías de desarrollo Web más utilizadas a la fecha. b. Se explicaron las principales características que poseen las metodologías de desarrollo Web seleccionadas en esta investigación. c. Se determinó y explicó la importancia y características de las herramientas CASE. d. Se determinaron y explicaron los criterios relacionados a los equipos de trabajo, que serán tomados en cuenta en el modelo de selección.

**Tabla 4.** Actividades/resultados por cada etapa planificada para la ejecución de la investigación

3. Análisis: se analizó toda la información recabada y se relacionaron las diferentes características investigadas para obtener artefactos (matrices), los cuales, serán la base teórica para el desarrollo del modelo, así como un cuestionario que guiará a los desarrolladores a seleccionar la metodología más apropiada.
4. Construcción del modelo y pruebas con algunos ejemplos: en esta fase se generó el modelo de selección de la metodología Web más apropiada según las características funcionales de la aplicación y el equipo de trabajo. Se realizan varios ejemplos para mostrar el funcionamiento del modelo generado.
5. Elaboración de conclusiones y recomendaciones: se expresan los resultados y recomendaciones de la investigación realizada para este trabajo de grado.

## **CAPÍTULO IV: Modelo de selección de la mejor metodología de desarrollo para una aplicación Web**

El objetivo de este capítulo es presentar el resultado del análisis y evaluación de la información recopilada, y el desarrollo del modelo propuesto en este trabajo de grado. La información utilizada para el análisis fue la siguiente:

- a. Proceso de desarrollo de las metodologías.
- b. Tipo de aplicaciones a desarrollar y el modelado requerido.
- c. Tipos de modelos utilizados para las metodologías.
- d. Herramientas CASE que utilizan las propuestas metodológicas estudiadas.

Con cada uno de estos aspectos se elaboraron cuadros (matrices) para mostrar el comportamiento de las propuestas metodológicas en ese ámbito. Las comparaciones y relaciones obtenidas se utilizaron para la creación del modelo de selección, que es el objetivo de este trabajo de grado.

El modelo, además cuenta con un cuestionario que facilitará a los diseñadores la selección de la metodología. En este capítulo también se presenta el diagrama del modelo, donde se explica de manera gráfica y escrita su funcionamiento.

#### IV.1 Relación de las fases del proceso de los métodos de modelado

METODOLOGÍAS	FASES	Fases del Ciclo de Vida de Sistemas					
		Levantamiento de Requerimientos	Análisis	Diseño	Desarrollo	Pruebas	Implementación
HDM	Authoring-in-the-large		■	▲			
	Authoring-in-the-small			■			
RMM	Crear Modelo E-R de la Aplicación		■	▲			
	Realizar el diseño de la presentación de cada entidad			■			
	Realizar el diseño de la navegación			■			
	Diseño del protocolo de conversión de los diagramas			■			
	Diseño de la Interfaz de Usuario			■			
	Diseño del comportamiento de la aplicación en la ejecución.			■			
	Construcción del sistema y pruebas				■	■	
EORM	Análisis del sistema		■				
	Creación del esquema EORM			■			
	Transformación del esquema EORM				■		

**Tabla 5.** Comparación de las fases de las metodologías de desarrollo Web con el ciclo de vida de desarrollo de Software

		Fases del Ciclo de Vida de Sistemas					
METODOLOGÍAS	FASES	Levantamiento de Requerimientos	Análisis	Diseño	Desarrollo	Pruebas	Implementación
OOHDM	Diseño Conceptual	■	■	■			
	Diseño Navegacional			■			
	Diseño de interfaces abstractas			■			
	Implementación				■		
OO-H	Diagrama de clases y diagrama de casos de usos	■	■				
	<i>Navigation Access Diagram (NAD).</i>			■			
	<i>Abstract Presentation Diagram (APD)</i>			■			
	<i>Apply refinements to APD</i>			■			
	<i>Generate Deliverables</i>				■		
SOHDM	Análisis del dominio	■	■				
	Modelo de objetos			■			
	Diseño de las vistas			■			
	Diseño Navegacional			■			
	Diseño de implementación			■	■		
	Construcción				■		

**Tabla 5.** Comparación de las fases de las metodologías de desarrollo Web con el ciclo de vida de desarrollo de Software (Continuación)

		Fases del Ciclo de Vida de Sistemas					
METODOLOGÍAS	FASES	Levantamiento de Requerimientos	Análisis	Diseño	Desarrollo	Pruebas	Implementación
WSDM	Modelo de Usuarios		■	■			
	Diseño Conceptual			■			
	Diseño de la implementación			■	■		
	Implementación				■		
WebML	Modelo Estructural	■	■				
	Modelo de Hipertexto			■			
	Modelo de Presentación			■			
	Modelo de Personalización			■			
UWE	Especificación de requerimientos	■					
	Modelo Lógico-Conceptual		■	■			
	Modelo de Navegación		■	■			
	Modelo de presentación		■	■			
OOWS	Especificación del problema	■	■	■			
	Desarrollo de la aplicación				■		

**Tabla 5.** Comparación de las fases de las metodologías de desarrollo Web con el ciclo de vida de desarrollo de Software (Continuación)

		Fases del Ciclo de Vida de Sistemas					
METODOLOGÍAS	FASES	Levantamiento de Requerimientos	Análisis	Diseño	Desarrollo	Pruebas	Implementación
W2000	Análisis de requerimientos	■	■				
	Diseño de hipermedia			■			
	Diseño de los servicios			■			
	Actividades de personalización			■			
NDT	Captura y definición de los requisitos	■					
	Análisis de la navegación		■	■			
	Diseño de la navegación			■			

**Tabla 5.** Comparación de las fases de las metodologías de desarrollo Web con el ciclo de vida de desarrollo de Software

Del análisis mostrado en la Tabla 5 se puede concluir que la mayoría de las propuestas metodológicas están enfocadas exclusivamente al análisis y diseño de la aplicación y ninguna toma en cuenta la fase de implementación del sistema, así como también se destaca que no solo HDM, EORM, WSDM y RMM, tienen una fase que permita recopilar y estudiar los requerimientos de la aplicación, lo que conlleva a que éstos métodos deben

ser completados con otros procesos que permitan la obtención de los requerimientos.

También destaca que solo RMM toma en cuenta la fase de prueba de la aplicación, característica que hace que equipos de trabajo que deseen tomar otra metodología diferente, deberán tener en cuenta otros métodos para que al concluir la fase de desarrollo (codificación) de una aplicación, puedan tener un proceso que permitan realizar pruebas validas al software resultante.

Todas las propuestas metodológicas para el desarrollo de aplicaciones Web, excepto NDT, WebML, UWE, W2000 y HDM toman en cuenta el proceso de codificación de la aplicación.

#### **IV.2 Tipos de aplicaciones Vs. tipos de modelado requerido**

- **Sitios Web centrados en la información:** En este tipo de aplicaciones se debe tomar en cuenta el modelado de la estructura del contenido y del hipertexto, ya que éstas aplicaciones se caracterizan por tener grandes volúmenes de contenido, los cuales no cambian de manera constante y la interacción con el usuario se limita a la navegación.
- **Aplicaciones Web interactivas:** Este tipo de aplicaciones se caracterizan por poseer un gran volumen de contenido heterogéneo, el

cual va evolucionando constantemente. Estas características hacen que en este tipo de aplicaciones se debe contar con el modelado de la estructura del contenido, del hipertexto y de la presentación del contenido. También en este tipo de aplicaciones se debe modelar el comportamiento del hipertexto y se debe personalizar el diseño para los tipos de usuarios que interactuarán con la aplicación.

- **Aplicaciones Web Transaccionales:** El diseño para este tipo de aplicaciones debe contener el modelado tanto de la estructura como del comportamiento del contenido, del hipertexto y de la presentación, así como también se debe personalizar el diseño para los tipos de usuarios que interactuarán con ella. También es relevante en este tipo de aplicaciones contar con un modelo que permita recopilar de manera precisa los requerimientos funcionales transaccionales y así definir correctamente los objetivos y el alcance.
- **Aplicaciones Web basadas en el flujo de trabajo:** El diseño de este tipo de aplicaciones se centra principalmente en el modelado de la estructura del contenido, del hipertexto y de la presentación, también se debe tomar en cuenta el comportamiento del hipertexto y de la presentación. La mayoría de este tipo de aplicaciones son dirigidas a apoyar el trabajo dentro de organizaciones, gestión de procesos y flujos. Es indispensable que el diseño sea personalizado según los

distintos tipos de usuarios que sean identificados. También es relevante contar con un modelo que permita recopilar de manera precisa los requerimientos funcionales y así definir correctamente los objetivos y el alcance de la aplicación.

- **Aplicaciones Web Colaborativas:** Estas aplicaciones se caracterizan principalmente por la distribución de la información. Es primordial que el diseño se centre en el modelado de la estructura, en el comportamiento del contenido y del hipertexto. También su diseño se centra en personalizar por tipo de usuarios la funcionalidad de la aplicación. Se debe hacer énfasis en el diseño de la presentación.
- **Web sociales:** Este tipo de aplicaciones está centrado en el modelado de la estructura y el comportamiento de contenido, del hipertexto y de presentación, así como también en personalizar por tipo de usuarios la funcionalidad de la aplicación. Además es relevante contar con un modelo que permita recopilar de manera precisa los requerimientos y así definir correctamente los objetivos y el alcance.
- **Portales Web:** Este tipo de aplicaciones están centradas en el modelado de la estructura y comportamiento del contenido, hipertexto y de presentación, así como también en personalizar por tipo de usuarios la funcionalidad de la aplicación. Igualmente, se debe contar con un modelo que permita recopilar de manera precisa los

requerimientos funcionales y así definir correctamente los objetivos y el alcance.

En la Tabla 6 se muestra un resumen de las necesidades de modelo según el tipo de aplicación.

<b>TIDO DE APLICACIÓN</b>	<b>Modelado de Requerimientos</b>	<b>Modelado de comportamiento Contenido</b>	<b>Modelado de comportamiento Hipertexto</b>	<b>Modelado de comportamiento Presentación</b>	<b>Modelado de Estructura Contenido</b>	<b>Modelado de Estructura Hipertexto</b>	<b>Modelado de Estructura Presentación</b>	<b>Modelado de Personalización</b>
<b>Sitios Web centrados en la información</b>								
<b>Aplicaciones Web interactivas</b>								
<b>Aplicaciones web Transaccionales</b>								
<b>Aplicaciones Web basadas en el flujo de trabajo</b>								
<b>Aplicaciones Web Colaborativas</b>								
<b>Web sociales</b>								
<b>Portales Web</b>								

**Tabla 6.** Tipo de aplicaciones Vs. tipo de modelados

### IV.3 Tipos de modelado que componen cada metodología de diseño Web

Según la revisión de las distintas propuestas metodológicas se observó que no todas proponen el modelado de los distintos elementos que caracterizan las aplicaciones Web. En todo desarrollo Web es fundamental realizar el modelado de la estructura del contenido y del hipertexto, este tipo de modelado se considera básico en todo diseño Web. En la Tabla 7 se muestran los tipos de modelado que llevan a cabo las propuestas metodológicas estudiadas.

METODOS DE MODELADO WEB	Modelado de Requerimientos	Modelado de comportamiento Contenido	Modelado de comportamiento Hipertexto	Modelado de comportamiento Presentación	Modelado de Estructura Contenido	Modelado de Estructura Hipertexto	Modelado de Estructura Presentación	Modelado de Personalización
HDM								
OO-H								
EORM								
OOHDM								
SOHDM								
OOWS								
RMM								
WebML								

**Tabla 7.** Tipos de modelados que contiene cada Metodologías de desarrollo Web estudiada en este trabajo de grado

<b>METODOS DE MODELADO WEB</b>	<b>Modelado de Requerimientos</b>	<b>Modelado de comportamiento Contenido</b>	<b>Modelado de comportamiento Hipertexto</b>	<b>Modelado de comportamiento Presentación</b>	<b>Modelado de Estructura Contenido</b>	<b>Modelado de Estructura Hipertexto</b>	<b>Modelado de Estructura Presentación</b>	<b>Modelado de Personalización</b>
<b>UWE</b>								
<b>W2000</b>								
<b>WSDM</b>								
<b>NDT</b>								

**Tabla 7.** Tipos de modelados que contiene cada Metodologías de desarrollo Web estudiada en este trabajo de grado

Como se puede observar en el cuadro anterior la propuesta metodológica HDM no toma en cuenta uno de estos dos tipos de modelos, lo cual hace que estos métodos tengan una gran deficiencia a la hora del diseño de la aplicación.

Las propuestas metodológicas como HDM, EORM, OOWS Y RMM no toman en cuenta, a la hora de diseñar, los tipos de usuarios que harán parte del entorno de la aplicación Web, lo que hace que la especificación del diseño sea de manera global y no personalizada.

EORM, WSDM y NDT no modelan la presentación de la aplicación, aspecto que no es totalmente indispensable, pero que si ayuda mucho a la

hora de la construcción de la aplicación y que facilita la aceptación y adaptación del usuario.

#### **IV.4 Reducción de los métodos de modelado Web.**

En la actualidad muchas de estas propuestas metodológicas no han evolucionado o simplemente su utilización en aplicaciones reales es muy escasa. Por lo tanto, para poder obtener un modelo de selección factible donde se elija la metodología para el desarrollo de una aplicación Web, es apropiado reducir la cantidad de métodos a seleccionar, para esto se van a tomar en cuenta tres factores:

1. Si el método cumple con herramientas básicas para el modelado de aplicaciones Web (modelado de la estructura de contenido y del hipertexto).
2. Si ha evolucionado con el pasar de los años, corrigiendo problemas detectados en el proceso de desarrollo propuesto.
3. El uso que han tenido las metodologías en el desarrollo de modelado de aplicaciones Web reales.

Tomando en cuenta los tres factores nombrados anteriormente, se decidió eliminar los siguientes Métodos:

- **HDM:** se decide apartar esta metodología por no establecer el modelado de la estructura del contenido y por no haber evolucionado con el tiempo.
- **EORM:** esta metodología es apartada por no haber evolucionado con el tiempo y por su poco uso en desarrollo de aplicaciones Web reales.
- **W2000:** se decide apartar esta metodología por su escaso uso en desarrollo de aplicaciones reales y por no haber evolucionado con el tiempo.
- **RMM:** se decide apartar esta propuesta metodológica por no haber evolucionado con el tiempo.
- **SOHDM:** esta metodología es apartada por no haber evolucionado con el tiempo y por su poco uso en desarrollo de aplicaciones Web reales.

Finalmente, para el modelo de selección del método de modelado de una aplicación Web, se tomarán en cuenta los siguientes métodos:

- |                |               |
|----------------|---------------|
| • <b>OO-H</b>  | • <b>UWE</b>  |
| • <b>OOHDM</b> | • <b>WSDM</b> |
| • <b>OOWS</b>  | • <b>NDT</b>  |
| • <b>WebML</b> |               |

## **IV.5 Análisis de las herramientas CASE de las metodologías de diseño**

### **Web**

En la actualidad muchos diseñadores prefieren metodologías que cuentan con una herramienta que los ayude en la documentación del diseño y que los guíe en el proceso de desarrollo de la aplicación. Por tal motivo, una vez hecha la selección de las metodologías a ser utilizadas por el modelo, se pasó a estudiar las características de las herramientas CASE que dan soporte a tales metodologías.

Como se expuso en el capítulo II, sección “herramientas CASE”, las propiedades que más interesan son:

- a. Integrado con las actividades de las metodología de diseño.
- b. Controles de coherencia del diseño de la aplicación.
- c. Manejo las versiones de los diseños realizados.
- d. Auto generación de parte de la interfaz y código de la aplicación.
- e. Auto generación de parte de la documentación de la aplicación

De la Tabla 2, se puede concluir que las metodologías OO-H y OOHDM son las que cuentan con las herramientas CASE más completas y funcionales. La metodología NDT cuenta con una herramienta CASE tan completa como la de OO-H y OOHDM, con la diferencia que sólo genera automáticamente el código de la interfaz (HTML) y obvia el código básico de

la aplicación. Las herramientas CASE con las que cuentan las metodologías UWE, OOWS y WebML sólo tienen la debilidad de no contar con un mecanismo que permita el respaldo de las versiones de los diseños de una aplicación, lo que podría causar un esfuerzo adicional en caso de que el diseñador quiera revertir los cambios realizados.

#### IV.6 Matriz de selección de la metodología de desarrollo Web.

A continuación, en la Tabla 8 se muestra la leyenda para interpretar lo presentado en la Tabla 9. En esta última se muestra la matriz donde se relacionan las metodologías de desarrollo Web, los tipos de modelados y los tipos de aplicaciones Web, además se muestra información relevante de cada una de las metodologías, como el lenguaje de modelado, el paradigma de modelado y el modelo de proceso que sigue. Esta matriz es la base fundamental del modelo de selección desarrollado en este trabajo de grado.

LEYENDA			
1	Modelado de Requerimientos	4	Modelado de Comportamiento Presentación
2	Modelado de Comportamiento Contenido	5	Modelado de Estructura Contenido
3	Modelado de Comportamiento Hipertexto	6	Modelado de Estructura Hipertexto
		7	Modelado de Estructura Presentación
		8	Modelado de Personalización

**Tabla 8.** Leyenda correspondiente a la Tabla 9.

PROPUESTAS METODOLÓGICAS WEB VS TIPO DE APLICACIONES WEB	Sitios Web centrados en la información (Tipos de Mod: 5,6)		Aplicaciones Web interactivas (Tipos de Mod: 3,5,6,7,8)		Aplicaciones Web transaccionales (Tipos de Mod: 1,2,3,4,5,6,7,8)		Aplicaciones Web basadas en el flujo de trabajo (Tipos de Mod: 1,3,4,5,6,7,8)	
	Aplica	NO Aplica	Aplica	NO Aplica	Aplica	NO Aplica	Aplica	NO Aplica
<b>OO-H</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,3,4,5,6,7,8 Mod. Proceso: Modelo Espiral / Interactivo - Incremental	5,6		3,5,6,7, 8		1,3,4,5, 6,7,8	2	1,3,4,5, 6,7,8	
<b>OOHDM</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML - Propia Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelado Incremental	5,6		3,5,6,7, 8		1,2,3,4, 5,6,7,8		1,3,4,5, 6,7,8	
<b>OOWS</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,4,5,6,7 Mod. Proceso: Modelo de Construcción de Prototipos	5,6		3,5,6,7	8	1,2,3,4, 5,6,7	8	1,3,4,5, 6,7	8
<b>UWE</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelo Espiral / Interactivo - Incremental	5,6		3,5,6,7, 8		1,2,3,4, 5,6,7,8		1,3,4,5, 6,7,8	
<b>WSDM</b> Paradigma de Mod.: HT Lenguaje de Mod.: Propia Tipos de Mod: 2,3,5,6,8 Mod. Proceso: Modelo Lineal Secuencial	5,6		3,5,6,8	7	2,3,5,6, 8	1,4,7	3,5,6,8	1,4,7
<b>WebML</b> Paradigma de Mod.: DB Lenguaje de Mod.: UML y E-R Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelo Interactivo	5,6		3,5,6,7, 8		1,2,3,4, 5,6,7,8		1,3,4,5, 6,7,8	
<b>NDT</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,5,6,8 Mod. Proceso: Modelo Lineal Secuencial	5,6		3,5,6,8	7	1,2,3,5, 6,8	4,7	1,3,5,6, 8	4,7

**Tabla 9.** Matriz de selección de la metodología de desarrollo Web, según el tipo de aplicación a realizar.

PROPUESTAS METODOLÓGICAS WEB VS TIPO DE APLICACIONES WEB	Aplicaciones Web colaborativas (Tipos de Mod.: 2,3,5,6,7,8)		Web Sociales (Tipos de Mod.: 1,2,3,4,5,6,7,8)		Portales Web (Tipos de Mod.: 1,2,3,4,5,6,7,8)	
	Aplica	NO Aplica	Aplica	NO Aplica	Aplica	NO Aplica
<b>OO-H</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,3,4,5,6,7,8 Mod. Proceso: Modelo Espiral / Interactivo - Incremental	3,5,6,7,8	2	1,3,4,5, 6,7,8	2	1,3,4,5, 6,7,8	2
<b>OOHDM</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML - Propia Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelado Incremental	2,3,5,6,8	7	1,2,3,4, 5,6,7,8		1,2,3,4, 5,6,7,8	
<b>OOWS</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,4,5,6,7 Mod. Proceso: Modelo de Construcción de Prototipos	2,3,5,6	8	1,2,3,4, 5,6,7	8	1,2,3,4, 5,6,7	8
<b>UWE</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelo Espiral / Interactivo - Incremental	2,3,5,6,7,8		1,2,3,4, 5,6,7,8		1,2,3,4, 5,6,7,8	
<b>WSDM</b> Paradigma de Mod.: HT Lenguaje de Mod.: Propia Tipos de Mod: 2,3,5,6,8 Mod. Proceso: Modelo Lineal Secuencial	2,3,5,6,8	7	2,3,5,6, 8	1,4,7	2,3,5,6, 8	1,4,7
<b>WebML</b> Paradigma de Mod.: DB Lenguaje de Mod.: UML y E-R Tipos de Mod: 1,2,3,4,5,6,7,8 Mod. Proceso: Modelo Interactivo	2,3,5,6,7,8		1,2,3,4, 5,6,7,8		1,2,3,4, 5,6,7,8	
<b>NDT</b> Paradigma de Mod.: OO Lenguaje de Mod.: UML Tipos de Mod: 1,2,3,5,6,8 Mod. Proceso: Modelo Lineal Secuencial	2,3,5,6,8	7	1,2,3,5, 6,8	4,7	1,2,3,5, 6,8	4,7

**Tabla 9.** Matriz de selección de la metodología de desarrollo Web, según el tipo de aplicación a realizar.

#### **IV.7 Prácticas en desarrollo Web.**

Para poder identificar que característica es más relevante a la hora de seleccionar una metodología de desarrollo Web, se hizo un análisis de varios estudios relacionados con las prácticas de desarrollo Web en distintas empresas de desarrollo.

El primer estudio revisado (Ramesh, Pries-Heje, & Baskerville, 2002) está enfocado en entender los rasgos que caracterizan los desarrollos Web. En este estudio estuvieron involucradas 9 compañías de desarrollo Web de Estados Unidos. El resultado del estudio fue la identificación de las 3 principales prácticas en el desarrollo Web:

- i. La gran velocidad con la que el mercado requiere la aplicación
- ii. La operación en diferentes tipos de mercados
- iii. La falta de experiencias en el desarrollo de los productos.

El segundo estudio revisado (McDonald & Welland, 2001) identifica practicas en el desarrollo Web similares al primer estudio revisado. En este estudio se descata la falta de experiencias en el desarrollo de los productos. En un estudio realizado en Singapur (Yusop, Zowghi, & Lowe, 2006), en donde se eligieron 5 proyectos de desarrollo Web realizado por compañías, y se analizó el impacto de los requerimientos no funcionales de los proyectos,

que muy poca veces son definidos o vagamente definidos, como por ejemplo los aspectos de seguridad y la integración con otros sistemas. En el estudio, entre otras cosas, consideraron:

- i. La capacidad del equipo de trabajo para atender los requerimientos no funcionales
- ii. La experiencia previa en las actividades involucradas.

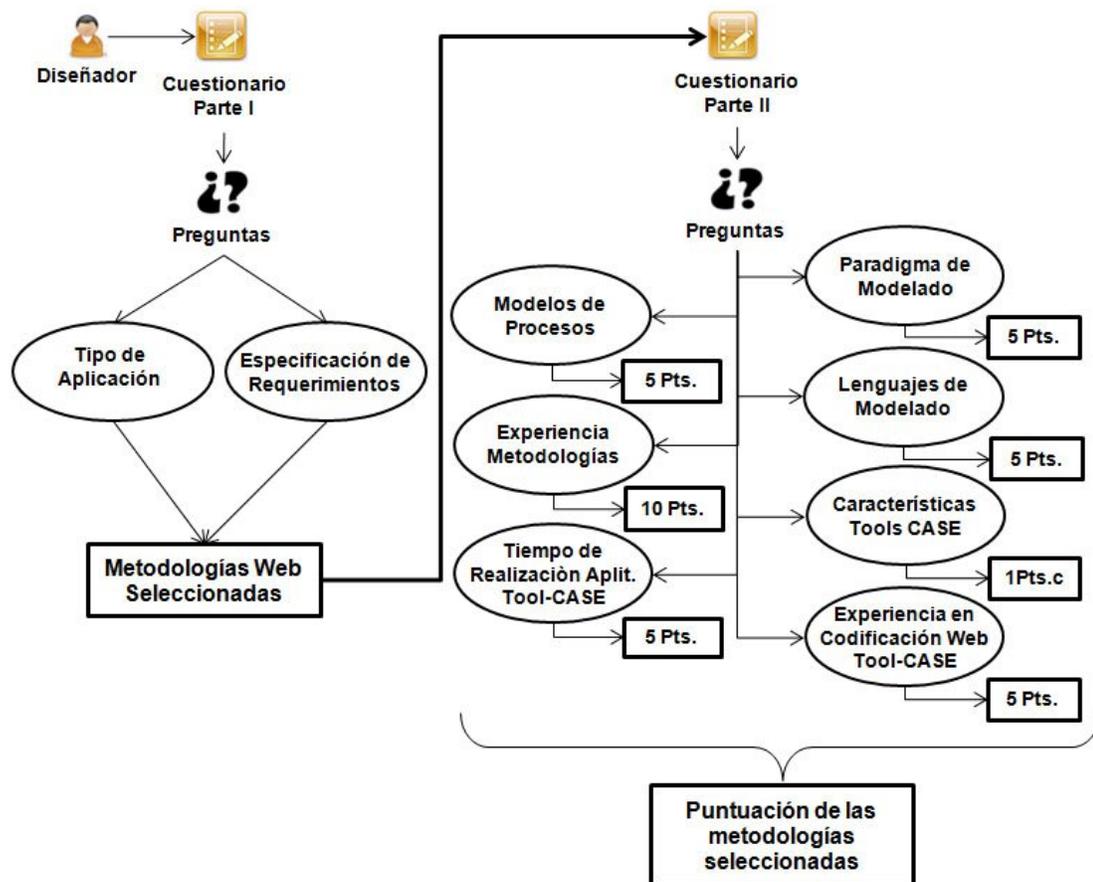
Una de las conclusiones fue que el equipo desarrollador es más eficaz en la medida que tenga mayor experiencia en desarrollo de aplicaciones Web.

Se puede concluir, que el conocimiento y la experiencia que posee el equipo de trabajo en el desarrollo de aplicaciones Web y en el uso de buenas prácticas influyen, significativamente, en el éxito del proyecto.

En base a lo expuesto, la experiencia del equipo desarrollador es un factor determinante para el éxito de la realización de una aplicación Web. Otros factores que influyen en el desarrollo son las herramientas que utilice el equipo de trabajo para llevar a cabo el proyecto, como lo son el uso de paradigmas de modelado, acceso a herramientas CASE, y la capacidad de instanciar modelos de procesos según las necesidades del desarrollo.

## IV.8 Modelo propuesto para la selección de la metodología de desarrollo Web de una aplicación según sus características funcionales

A continuación, en la Figura N° 6, se presenta el modelo desarrollado en este trabajo de grado para la selección de la mejor metodología de desarrollo para la realización de una aplicación Web.



**Figura N° 6.** Modelo para la selección de la metodología de desarrollo Web de una aplicación según sus características funcionales

El modelo comienza cuando el diseñador completa la Parte I del cuestionario (Apéndice 1), dando como resultado la lista de las metodologías que cumple con los modelados requeridos para el tipo de aplicación a realizar (Tabla 8 y 9) y también si el diseñador ya cuenta con todos los requerimientos o debe empezar desde cero, ya que para eso se necesita una metodología que contemple una fase de levantamiento de requerimientos (Tabla 5).

Posteriormente se procede a responder la Parte II del cuestionario, donde se le asignará un puntaje a cada metodología seleccionada según la respuesta de cada pregunta. A continuación se describe cómo será el puntaje de cada pregunta:

- Pregunta N° 1. Modelo de proceso: es una pregunta de selección múltiple, las metodologías que coincidan con alguna de la(s) respuesta(s) (Tabla 8 y 9) se le dará un valor de 5 puntos.
- Pregunta N° 2. Paradigma de modelado: es una pregunta de selección múltiple, las metodologías que coincidan con alguna de la(s) respuesta(s) (Tabla 8 y 9) se le dará un valor de 5 puntos.
- Pregunta N° 3. Lenguaje de modelado: es una pregunta de selección múltiple, las metodologías que coincidan con alguna de la(s) respuesta(s) (Tabla 8 y 9) se le dará un valor de 5 puntos.

- Pregunta N° 4. Experiencia en metodologías Web: es una pregunta de selección múltiple, las metodologías que coincidan con alguna de la(s) respuesta(s) se le otorgara un puntaje de 10.
- Pregunta N° 5. Características herramientas CASE: es una pregunta de selección múltiple, donde se le otorgará 1 punto por cada característica seleccionada que coincida con las herramientas CASE de las metodologías seleccionadas (Tabla 2).
- Pregunta N° 6. Tiempo de realización de la aplicación: es una pregunta de selección única, en la que se otorgará un valor de 5 puntos si la respuesta es “ajustada” o “muy ajustada” y la metodología cuente con una herramienta CASE que genere automáticamente la interfaz y el código de la aplicación (Tabla 2).
- Pregunta N° 7. Experiencia en codificación Web: es una pregunta de selección única, se le otorgará un valor de 5 puntos si la respuesta es “NO” y la metodología cuente con una herramienta CASE que genere automáticamente la interfaz y el código de la aplicación (Tabla 2).

Casi todas las preguntas de la Parte II del cuestionario tienen un puntaje máximo de 5 puntos. Se hace la excepción para la pregunta N° 4, pues en base a las buenas prácticas, el hecho de conocer y tener experiencia en metodologías de desarrollo Web es factor determinante para el éxito del

proyecto. Además las metodologías incluyen paradigmas de modelos, notación y posibilidad de instanciar modelos de procesos. Esta relación se mostro en la Tabla 9.

En el siguiente capítulo se mostrarán 3 ejemplos de la utilización del modelo propuesto.

## **CAPÍTULO V: Ejemplos del funcionamiento del modelo de selección de la mejor metodología de desarrollo para una aplicación Web**

En este capítulo se desarrollarán tres ejemplos en donde se podrá observar como es el funcionamiento del modelo de selección y como va cambiando el resultado de su aplicación según cambian los parámetros de entrada.

### **V.1 Ejemplo N° 1.**

Un equipo de trabajo tiene la responsabilidad de diseñar y desarrollar una aplicación Web con la siguiente característica:

- Una aplicación de venta de artículos deportivos.
- Debe manejar inventario.
- Maneja usuarios (compradores).
- Se tiene claro el alcance y objetivos de la aplicación.

Otras de las características a tomar en cuenta son las siguientes:

- Tiempo de entrega de la aplicación es de 4 meses.
- Equipo con poca experiencia en el diseño de aplicaciones Web, pero tiene experiencia en el uso de lenguaje de programación para Web.
- El equipo de trabajo está conformado por dos programadores y un líder de proyecto. Todos van a participar en el diseño y desarrollo de la aplicación.

- Experiencia en el modelado OO y en el lenguaje de modelado UML.
- Experiencia en los modelos de proceso lineal secuencial, prototipo, incremental y espiral

Si a este caso le aplicamos el cuestionario se tendrían las siguientes respuestas:

<b>Cuestionario</b>	<b>Pregunta / Respuesta</b>
<b>Cuestionario Parte I</b>	<ol style="list-style-type: none"> <li>1. Aplicaciones Web Transaccionales</li> <li>2. Si</li> </ol>
<b>Cuestionario Parte II</b>	<ol style="list-style-type: none"> <li>1. El modelo de construcción de prototipos, el modelo DRA, el modelo espiral y el modelo de desarrollo concurrente</li> <li>2. O-O (Modelo Orientado a Objetos)</li> <li>3. UML</li> <li>4. N/A</li> <li>5. Que esté integrado con las actividades de las metodología de diseño, que genere parte de la interfaz y código de la aplicación y que genere parte de la documentación de la aplicación</li> <li>6. Ajustado</li> <li>7. Si</li> </ol>

**Tabla 10.** Solución del cuestionario para el ejemplo N°1

## ¿Cómo Reacciona el Modelo?

Aplicando el modelo de selección (Figura N° 6), de la primera parte del cuestionario se obtiene que las metodologías seleccionadas son: OOHDM, UWE y WebML, ya que son las únicas que contemplan los modelados de diseño requeridos para este tipo de aplicación (Tabla 8 y 9). Posteriormente, en la segunda parte del cuestionario se le asigna la siguiente puntuación a las metodologías seleccionadas:

- Pregunta N° 1: UWE es la única de las tres metodologías en donde coincide con alguna de las respuestas (Ver Tabla 8 y 9), por lo tanto UWE recibe 5 puntos y las otras dos 0 puntos.
- Pregunta N° 2: UWE y OOHDM son metodologías donde su paradigma de modelado es O-O, cambio WebML contempla un paradigma diferente (Ver Tabla 8 y 9), por lo tanto UWE y OOHDM reciben 5 puntos y WebML 0 puntos.
- Pregunta N°3: UWE, OOHDM y WebML son metodologías que utilizan UML como su lenguaje de modelado (Ver Tabla 8 y 9), por lo tanto las tres metodologías reciben 5 puntos.
- Pregunta N° 4: como la respuesta fue N/A las tres metodologías reciben un puntaje de 0 por esta pregunta.
- Pregunta N° 5: las tres metodologías seleccionadas poseen herramientas CASE que contemplan las tres características

seleccionadas (Ver Tabla 2), por lo tanto cada metodología recibe 3 puntos, uno por cada característica que coincide.

- Pregunta N° 6: como la respuesta es “ajustada” y las metodologías UWE y WebML cuentan con una herramienta CASE que genera automáticamente la interfaz y el código de la aplicación, se le otorga a cada una de éstas metodologías un puntaje de 5, mientras a la metodología OOHDM se le otorga 0 puntos.
- Pregunta N° 7: como la respuesta es “Si”, esta pregunta no otorgará puntos a ninguna de las metodologías Web seleccionadas.

A continuación se muestra el cuadro de puntaje final y el resultado, de aplicar el modelo de selección:

Métodos	Puntos Obtenidos en Cada Pregunta							TOTAL
	Preg. 1	Preg. 2	Preg. 3	Preg. 4	Preg. 5	Preg. 6	Preg. 7	
OOHDM	0	5	5	0	3	5	0	18
UWE	5	5	5	0	3	5	0	23
WebML	0	0	5	0	3	5	0	13

**Tabla 11.** Resultados de la aplicación del modelo de selección para el ejemplo N° 1

La metodología selecciona como la más adecuada para llevar a cabo la aplicación Web propuesta en el ejemplo N° 1 para el equipo de trabajo con las características mencionadas anteriormente es: **UWE**.

## V.2 Ejemplo N° 2.

Es el mismo caso que el ejemplo N° 1 pero con las siguientes condiciones del desarrollo y del equipo de trabajo. Se resalta en el ejemplo aquella condición que lo diferencia.

- Tiempo de entrega de la aplicación es de 6 meses. El tiempo de entrega es mucho más holgado que el ejemplo N°1.
- Equipo con experiencia en el diseño de aplicaciones Web.
- El equipo de trabajo está conformado por dos programadores y un líder de proyecto, donde todos van a participar en el diseño y desarrollo de la aplicación.
- Experiencia en el modelado y nomenclatura E-R. En el ejemplo N°1 la experiencia del equipo de trabajo es en el modelado OO y en el lenguaje de modelado UML

Si a este caso le aplicamos el cuestionario se tendrían las siguientes respuestas:

Cuestionario	Pregunta / Respuesta
<b>Cuestionario Parte I</b>	<ol style="list-style-type: none"> <li>1. Aplicaciones Web Transaccionales.</li> <li>2. Si.</li> </ol>
<b>Cuestionario Parte II</b>	<ol style="list-style-type: none"> <li>1. El modelo lineal secuencial, el modelo de construcción de prototipos, y el modelo espiral.</li> <li>2. DB (Modelo Entidad-Relación)</li> <li>3. E-R (Entidad – Interrelación)</li> <li>4. N/A</li> <li>5. Que esté integrado con las actividades de la metodología de diseño, que tenga controles de coherencia del diseño de la aplicación y que genere parte de la documentación de la aplicación.</li> <li>6. Holgado.</li> <li>7. Si.</li> </ol>

**Tabla 12.** Solución del cuestionario para el ejemplo N°2

### **¿Cómo reacciona el modelo?**

Aplicando el modelo de selección (Figura N° 6), de la primera parte del cuestionario se obtiene que las metodologías seleccionadas son: OOHDM, UWE y WebML, ya que son las únicas que contemplan los modelados de diseño requeridos para este tipo de aplicación (Tabla 8 y 9). Posteriormente de la segunda parte del cuestionario se le asigna la siguiente puntuación a las metodologías seleccionadas:

- Pregunta N° 1: UWE es la única de las tres metodologías en donde coincide con alguna de las respuestas (Ver Tabla 8 y 9), por lo tanto UWE recibe 5 puntos y las otras dos 0 puntos.
- Pregunta N° 2: UWE y OOHDM son metodologías donde su paradigma de modelado es O-O, en cambio WebML tiene como paradigma de modelado DB (Ver Tabla 8 y 9), por lo tanto WebML reciben 5 puntos y UWE y OOHDM 0 puntos.
- Pregunta N° 3: WebML es la única metodología, de las tres seleccionadas, que utilizan E-R como su lenguaje de modelado (Ver Tabla 8 y 9), por lo tanto recibe 5 puntos.
- Pregunta N° 4: como la respuesta fue N/A las tres metodologías reciben un puntaje de 0 por esta pregunta.
- Pregunta N° 5: las tres metodologías seleccionadas poseen herramientas CASE que contemplan las tres características seleccionadas (Ver Tabla 2), por lo tanto cada metodología recibe 3 puntos, uno por cada característica que coincide.
- Pregunta N° 6: como la respuesta es “Holgada” las tres metodologías reciben un puntaje de 0 por esta pregunta.
- Pregunta N° 7: como la respuesta es “Sí”, esta pregunta no otorga puntos a ninguna de las metodologías Web seleccionadas.

A continuación se muestra el cuadro de puntaje final y el resultado de aplicar el modelo de selección:

Métodos	Puntos Obtenidos en Cada Pregunta							TOTAL
	Preg. 1	Preg. 2	Preg. 3	Preg. 4	Preg. 5	Preg. 6	Preg. 7	
OOHDM	0	0	0	0	3	0	0	3
UWE	5	0	0	0	3	0	0	8
WebML	0	5	5	0	3	0	0	13

**Tabla 13.** Resultados de la aplicación del modelo de selección para el ejemplo N° 2

La metodología selecciona como la más adecuada para llevar a cabo la aplicación Web propuesta en el ejemplo N° 2 para el equipo de trabajo con las características mencionadas anteriormente es: **WebML**.

### V.3 Ejemplo N° 3.

Es el mismo caso que el ejemplo N°1 pero con las siguientes características del equipo de trabajo. Se resalta en el ejemplo aquella condición que lo diferencia.

- Tiempo de entrega de la aplicación es de tres meses. El tiempo es aún más ajustado que en el ejemplo N°1.
- Equipo con poca experiencia en el diseño de aplicaciones Web.

- El equipo de trabajo está conformado por dos programadores y un líder de proyecto, donde todos van a participar en el diseño y desarrollo de la aplicación.
- Experiencia en el modelado OO y nomenclatura UML.
- Experiencia en el modelado de proceso Incremental y espiral. En el ejemplo N°1 la experiencia del equipo de trabajo en el modelado de proceso es: lineal secuencial, prototipo, incremental y espiral.
- Consideran fundamental contar con una herramienta CASE que los ayude a construir la documentación y un prototipo, así como también los guíe en el proceso y permita guardar versiones. En el ejemplo N°1 el equipo de trabajo no determina las características que le gustaría que incluyera una herramienta CASE.

Si a este caso le aplicamos el cuestionario se tendrían las siguientes respuestas:

Cuestionario	Pregunta / Respuesta
<b>Cuestionario Parte I</b>	<ol style="list-style-type: none"> <li>1. Aplicaciones Web Transaccionales.</li> <li>2. Si.</li> </ol>
<b>Cuestionario Parte II</b>	<ol style="list-style-type: none"> <li>1. El modelo incremental y el modelo espiral.</li> <li>2. O-O (Modelo Orientado a Objetos)</li> <li>3. UML.</li> <li>4. N/A.</li> <li>5. Que esté integrado con las actividades de las metodologías de diseño, que tenga controles de coherencia del diseño de la aplicación, que maneje las versiones de los diseños realizados, que genere parte de la interfaz y código de la aplicación y que genere parte de la documentación de la aplicación.</li> <li>6. Ajustado.</li> <li>7. No.</li> </ol>

**Tabla 14.** Solución del cuestionario para el ejemplo N°3

### ¿Cómo reacciona el modelo?

Aplicando el modelo de selección (Figura N° 6), De la primera parte del cuestionario, se obtiene que las metodologías seleccionadas son: OOHD, UWE y WebML, ya que son las únicas que contemplan los modelados de diseño requeridos para este tipo de aplicación (Tabla 8 y 9). Posterior a la segunda parte del cuestionario se le asigna la siguiente puntuación a las metodologías seleccionadas:

- Pregunta N° 1: UWE Y OOHDM son las metodologías que coinciden con alguna de las respuestas (Ver Tabla 8 y 9), por lo tanto reciben 5 puntos y WebML 0 puntos.
- Pregunta N° 2: UWE y OOHDM son metodologías donde su paradigma de modelado es O-O, cambio WebML, contempla un paradigma diferente (Ver Tabla 8 y 9), por lo tanto UWE y OOHDM reciben 5 puntos y WebML 0 puntos.
- Pregunta N°3: UWE, OOHDM y WebML son metodologías que utilizan UML como su lenguaje de modelado (Ver Tabla 8 y 9), por lo tanto las tres metodologías reciben 5 puntos.
- Pregunta N° 4: como la respuesta fue N/A, las tres metodologías reciben un puntaje de 0 por esta pregunta.
- Pregunta N° 5: OOHDM es la única metodología de las seleccionadas que posee una herramientas CASE que contemplan las cinco características seleccionadas (Ver Tabla 2), por lo tanto recibe 5 puntos, en cambio las herramientas CASE de UWE y WebML contemplan cuatro de las cinco características seleccionadas, en consecuencia reciben 4 puntos cada una.
- Pregunta N° 6: como la respuesta es “ajustada” y las metodologías UWE y WebML cuentan con una herramienta CASE que genera automáticamente la interfaz y el código de la aplicación, se le otorga a

cada una de estas metodologías un puntaje de 5, mientras a la metodología OOHDM se le otorgan 0 puntos.

- Pregunta N° 7: como la respuesta es “No” y las tres metodologías seleccionadas poseen herramientas CASE que contemplan la generación de parte de la interfaz y del código de la aplicación, cada una de ellas recibe 5 puntos.

A continuación se muestra el cuadro de puntaje final y el resultado de aplicar el modelo de selección:

Métodos	Puntos Obtenidos en Cada Pregunta							TOTAL
	Preg. 1	Preg. 2	Preg. 3	Preg. 4	Preg. 5	Preg. 6	Preg. 7	
OOHDM	5	5	5	0	5	5	5	30
UWE	5	5	5	0	4	5	5	29
WebML	0	0	5	0	4	5	5	19

**Tabla 15.** Resultados de la aplicación del modelo de selección para el ejemplo N° 3

La metodología selecciona como la más adecuada para llevar a cabo la aplicación Web propuesta en el ejemplo N° 3 para el equipo de trabajo con las características mencionadas anteriormente es: **OOHDM**.

## **CAPÍTULO VI: Conclusiones y recomendaciones**

En este capítulo se presentan los aspectos más relevantes que se observaron y estudiaron durante el desarrollo del modelo propuesto en este trabajo de grado, así como las recomendaciones para trabajos futuros.

El resultado de este trabajo de grado fue la creación de un modelo que permita a los diseñadores de aplicaciones Web, seleccionar la metodología adecuada para llevar a cabo un proyecto, tomando en cuenta la naturaleza de la aplicación a desarrollar y la experiencia de los integrantes del equipo de trabajo.

Por lo investigado, analizado y desarrollado, se puede concluir que los factores más relevantes a la hora de seleccionar la metodología para llevar a cabo el desarrollo de una aplicación Web son los siguientes:

- **Tipo de aplicación a desarrollar:** el diseño fundamental de toda aplicación Web debe contener, obligatoriamente, el diseño de la estructura del contenido y del hipertexto, y según la naturaleza de la aplicación que se desea desarrollar, se pueden necesitar otros tipos de modelados. Es fundamental que la metodología que se vaya a utilizar para su diseño, provea los métodos para desarrollar cada uno de los modelados requeridos.

- **Paradigma de modelado:** una aplicación puede ser diseñada desde diferentes puntos de vista, como por ejemplo orientada a objeto (OO). Esto permitiría que el equipo de desarrollo se sienta más cómodo diseñando una aplicación, si la metodología que está utilizando se basa en algún paradigma de modelado en el cual tengan experiencia.
- **Modelo de proceso:** Todas las metodologías poseen un modelo de proceso, es decir, como se va a llevar a cabo la secuencia de las actividades propuestas. Cuando un equipo de trabajo tiene experiencia en el mismo modelo de proceso de la metodología seleccionada, se sienten más cómodos con la forma en que se van realizando las actividades.
- **Lenguaje de modelado:** es la notación, artefactos, símbolos y reglas que se utilizan para expresar los modelos que representan la aplicación. Por lo tanto, el proceso de diseño sería más rápido y fiable si el desarrollador conoce el o los lenguajes de modelado que se usan en la metodología seleccionada, ya que tendría un conocimiento previo de las herramientas utilizadas por la metodología en cada una de sus fases.
- **Tiempo establecido para el desarrollo de la aplicación:** en muchas ocasiones el tiempo con el que cuenta un equipo de trabajo para llevar a cabo una aplicación Web es ajustado, por lo que en estos casos

sería de gran utilidad para el equipo de trabajo, que la metodología que estén utilizando cuente con una herramienta que les facilite la realización de las actividades y que les ahorre tiempo en todo lo referente a la realización de la documentación del aplicativo y el código del diseño de la interfaz (HTML).

- **Experiencia en la tecnología para la implantación de aplicaciones**

**Web:** cuando un equipo de trabajo no tiene experiencia en la tecnología a utilizar en la programación de aplicaciones Web, es de gran utilidad que la metodología que estén usando contara con una herramienta que les genere parte del código básico de la aplicación, ya que esto los ayudaría en el conocimiento y aprendizaje de la nueva tecnología.

Contar con una metodología adecuada a la hora de desarrollar una aplicación Web, ayudaría a disminuir los problemas de diseño que han venido presentando este tipo de aplicaciones, reduciendo sus costos de desarrollo e implementación, haciendo que los objetivos planteados para el proyecto se cumplan de manera rápida y efectiva.

El desarrollo de aplicaciones efectivas permite disminuir el rechazo que, normalmente, tienen los usuarios a la hora de aceptar una nueva aplicación. En el caso de aplicaciones Web, este rechazo puede ser más evidente, ya que es una tecnología relativamente nueva y con características

de uso y navegación totalmente diferentes a las aplicaciones tradicionales, lo que hace que la utilización de una metodología de desarrollo adecuada para llevar a cabo una aplicación Web sea mucho más relevante.

Hay que destacar que cada vez son más comunes las aplicaciones Web dentro de las organizaciones, inclusive muchas aplicaciones tradicionales han sido migradas a la tecnología Web, por las bondades que esta ofrece. Se prevé que a mediano plazo las aplicaciones Web desplacen a las aplicaciones tradicionales, una muestra de esto son compañías como Microsoft y Google, que se encuentran trabajando en el primer sistema operativo Web, lo que producirá una nueva era en el desarrollo de software, en donde la ingeniería Web será la protagonista en esta nueva etapa.

Por lo expuesto, es fundamental que se realicen investigaciones relacionadas con el área de la ingeniería Web, que aporten conocimientos y experiencias, con la finalidad de contar, cada vez más, con una sólida base teórica que ayude a diseñar y desarrollar aplicaciones Web, y que faciliten la tarea de los equipos de trabajo en la transición de la migración de software tradicional al ambiente Web.

En este sentido, la finalidad de este trabajo de grado, fue aportar un enfoque novedoso en el mundo de la ingeniería Web y a su vez incentivar a nuevos investigadores del área, así como también formar parte del material de referencia obligada en futuras investigaciones relacionadas con el tema.

Sin duda, la funcionalidad del modelo desarrollado en este trabajo, podrá ser comprobada y puesta en práctica dentro del mundo laboral. Para ello se podría utilizar el método de DESMET (Kitchenham, 1996), el cual propone una cantidad de métodos para probar modelos dentro de organizaciones.

## **CAPÍTULO VII: Referencias Bibliográficas.**

**Baresi, L., Garzotto, F., & Maritati, M. (2002).** W2000 as a MOF Metamodel. 6th World Multiconference on Systemics, Cybernetics and Informatics - Web Engineering track, 1. Orlando.

**Baskerville, R. (1999).** Investigating Information Systems with Action Research. Communications of The Association for Information Systems , 2 (19), 1-32.

**Bianchini, A. (2000).** Modelo Referencial de Hipermedio, basado en teoría de grafos, para minimizar el problema de desorientación del usuario. Congreso Internacional de Computación CIC 2000, (págs. 121-130). Mexico.

**Booch, G., Rumbaugh, J., & Jacobson, I. (2006).** El lenguaje unificado de modelado 2ED. Massachusetts, USA: Addison Wesley.

**Cachero, C. (2003).** OO-H: Una extensión a los métodos OO para el modelado y generación automática de interfaces hipermediales. Tesis Doctoral, Universidad de Alicante, España.

**Cachero, C., & Gómez, J. (2003).** OO-H Method: extending UML to model web interfaces. En Information modeling for internet applications (págs. 144-173). Hershey: IGI Publishing.

**Campbell, B., & Goodman, J. (1988).** HAM: a general purpose hypertext abstract machine. Communications of the ACM. 31, págs. 856-861. New York: ACM.

**Ceri, S., Fraternali, P., & Bongio, A. (2000).** Web Modeling Language (WebML): a modeling language for designing Web sites. *Computer Networks: The International Journal of Computer and Telecommunications Networking*. 33, págs. 137-157. New York: Elsevier North-Holland, Inc.

**Conallen, J. (2002).** Building Web Applications with Uml, 2nd edition. Boston: Addison-Wesley Longman Publishing Co., Inc.

**Cutter. (7 de Noviembre de 2000).** POOR PROJECT MANAGEMENT NUMBER-ONE PROBLEM OF OUTSOURCED E-PROJECTS. Recuperado el 2009, de <http://www.cutter.com/research/2000/crb001107.html>.

**De Bra, P., Houben, G., & Kornatzky, Y. (1993).** An extensible data model for hyperdocuments. *ACM conference on Hypertext* (págs. 222-231). New York: ACM.

**De Troyer, O., & Leune, C. (1998).** WSDM: a user-centered design method for. *Computer networks and ISDN systems, 7th international world wide web conference*, (págs. 85–94). Elsevier.

**Deshpande, Y., Murugesan, S., Hansen, S., & Ginige, A. (2001).** Web Engineering : A New Discipline for Development of Web-Based Systems. *Lecture Notes in Computer Science*. 2016, págs. 3-13. Londres: Springer-Verlag.

**Diaz, P., & Aedo, I. (2007).** Towards efficient web engineering approaches through flexible process models. *Journal of Systems and Software*, 80, págs. 1375-1389.

**Escalona, M., Mejias, M., Torres, J., & Reina, A. (2002).** Desarrollo de la navegación en entornos web. II Taller en Ingeniería del Software Orientada al Web. Madrid.

**Eyssautie, M. (2006).** Metodología de la investigación. Desarrollo de la inteligencia (5 ed.). Mexico: Thomson Learning.

**Fons, J., Pastor, O., Valderas, P., & Ruiz, M. (2002).** OOWS: Un Método de Producción de Software en Ambientes Web. European E-Commerce Workshop, 33. Valencia - España.

**Fraternali, P. (1999).** Tools and Approaches for Data-intensive Web Applications. ACM Computing Surveys, 31, págs. 227-263.

**Furura, R., & Stotts, P. (1991).** A functional meta-structure for hypertext models and systems. Technical Report. College Park: University of Maryland.

**Furuta, R., & Stotts, P. (1989).** Petri-net-based hypertext: document structure with browsing semantics. ACM Transactions on Information Systems (TOIS). 7, págs. 3-29. New York: ACM.

**Garzotto, F., Paolini, P., & Schwabe, D. (1993).** HDM - A model-based approach to hypermedia applications design. ACM Transactions on Information Systems, 11, págs. 1-23.

**Halasz, F., & Schwartz, M. (1994).** The Dexter hypertext reference model. Communications of the ACM. 37, págs. 30-39. New York: ACM.

**Hardman, L., Bulterman, D., & Van Rossum, G. (1994).** Amsterdam Hypermedia Model: extending hypertext to support real multimedia. Communications of the ACM. 37, págs. 50-62. New York: ACM.

**Instituto Superior Politécnico José Antonio Echeverría (1982).** Arquitectura y urbanismo vol. 4 . Universidad de Texas, USA.

**Isakowitz, T., Stohr, E., & Balasubramanian, P. (1995).** RMM: A methodology for the Design of Structured Hypermedia Applications. Communications of the ACM. 38, págs. 34–44. New York: ACM.

**Kappel, G., Pröll, B., Reich, S., & Retschitzegger, W. (2003).** Web Engineering. Munich: John Wiley & Sons Ltd.

**Kendall, K., & Kendall, J. (2005).** Análisis Y Diseño De Sistemas - 6ª Ed. Mexico: Prentice Hall.

**Kitchenham, B. (1996).** “DESMET: A method for evaluating Software Engineering methods and tools”. Staffordshire, U.K.

**Koch, N., & Kraus, A. (2002).** The expressive Power of UML-based Web Engineering. Second International Workshop on Web-oriented Software Technology, (págs. 105-119). Malaga.

**Lange, D. (1990).** A Formal Model of Hypertext. NIST Hypertext Standardization Workshop, (págs. 145-166). Washington.

**Lange, D. (1996).** An object-oriented design approach for developing hypermedia information systems. Journal of Organizational Computing and Electronic Commerce. 6, págs. 269-293. Hillsdale: L. Erlbaum Associates Inc.

**Lee, H., Lee, C., & Yoo, C. (1998).** A Scenario-based Object-Oriented Hypermedia Design Methodology for Developing Hypermedia Information Systems. 31th Hawaii International Conference on System Sciences, 2, págs. 47-56. Los Alamitos.

**Markovic, I., & Pereira, A. (2007).** Towards a formal framework for reuse in business process modeling. 5th International Conference on Business Process Management, (págs. 484-495). Australia.

**Martin, J. (1990).** Hyperdocuments & How to Create Them. New Jersey, USA: Prentice – Hall.

**McDonald, A., & Welland, R. (2001).** Agile Web Engineering (AWE) Process. Technical Report TR-2001-98. Glasgow.

**Pastor, J., & Saorín, T. (1998).** Cuadernos de Documentación Multimedia. Universidad Complutense de Madrid, 7. Madrid, España.

**Pastor, O., Insfrán, E., & Pelechano, V. (1999).** OO-Method: an software production environment. 9th Conference on Advanced Information System Engineering, (págs. 145 - 158). Barcelona, España.

**Piattini, M. (2007).** Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión. Madrid, España: Ra-ma.

**Pressman, R. (2005).** Ingeniería del Software (5 ed.). Madrid, España: McGrawHill.

**Ramesh, R., Pries-Heje, J., & Baskerville, R. (2002).** Annals of Software Engineering. Internet Software Engineering: A Different Class of Processes, 14, págs. 169-195.

**Retschitzegger, W., & Schwinger, W. (2000).** Towards Modeling of DataWeb Applications: A Requirement's Perspective. Americas Conference on Information Systems (AMCIS). Long Beach.

**Schwabe, D., & Rossi, G. (2000).** An Object Oriented Approach to Web-Based Application Design. Theory and Practice of Object Systems, 4, págs. 207-225.

**Schwabe, D., & Rossi, G. (1995).** The Object-Oriented Hypermedia Design Model (OOHDM). Communications of the ACM, 35, págs. 45-46.

**Selmi, S., Kraïem, N., Ghézala, B., & Hajjami, H. (2008).** Guidance in Web Applications Design. The International Workshop on Model Driven Information Systems Engineering: Enterprise, User and System Models, (págs. 114-125). Montpellier, France.

**Sommerville, I. (2002).** Ingeniería de Software (7 ed.). Madrid, España: Prentice Hall.

**Yusop, N., Zowghi, D., & Lowe, D. (2006).** The impacts of nonfunctional. Proceeding of European and Mediterranean Conference on Information Systems. Alicante.

**Zakon, R. (2010).** zakon.org. Recuperado el 02 de 03 de 2010, de zakon.org: <http://www.zakon.org/robert/internet/timeline/>

## **APÉNDICE 1**

A continuación se muestra el cuestionario que se propondrá a los diseñadores y cuyas respuestas serán la información que alimentará el modelo de selección.

### **Parte I. Selección de las metodologías que se ajusten al diseño requerido de la aplicación Web a desarrollar**

1. ¿Qué tipo de aplicación Web piensa desarrollar?
  - a. Sitios Web centrados en la información.
  - b. Aplicaciones Web interactivas.
  - c. Aplicaciones Web transaccionales.
  - d. Aplicaciones Web basadas en el flujo de trabajo.
  - e. Aplicaciones Web colaborativas.
  - f. Web sociales.
  - g. Portales Web.
  
2. ¿El equipo de trabajo cuenta con la especificación completa de los requerimientos de la aplicación a desarrollar?
  - a. Si
  - b. No

## Parte II. Información sobre la experiencia del equipo de desarrollo

1. Seleccione en cuál de los siguientes modelos de procesos tiene alguna experiencia:
  - a. El modelo lineal secuencial.
  - b. El modelo de construcción de prototipos.
  - c. El modelo DRA.
  - d. El modelo incremental.
  - e. El modelo espiral.
  - f. El modelo de desarrollo concurrente.
  - g. Desarrollo basado en componentes.
  - h. El modelo de métodos formales.
  
2. ¿En cuál de los siguientes paradigmas de modelado posee más experiencia?
  - a. DB (Modelo Entidad-Relación)
  - b. O-O (Modelo Orientado a Objetos)
  - c. H-T (Modelo Orientado a Hipertexto)

3. ¿En cuál de los siguientes lenguajes de modelado posee más experiencia?
  - a. E-R (Entidad Interrelación)
  - b. UML
  - c. Otro
  
4. ¿El equipo de trabajo tiene experiencia en alguna de las siguientes metodologías?
  - a. OO-H
  - b. OOHDM
  - c. OOWS
  - d. WebML
  - e. UWE
  - f. WSDM
  - g. NDT
  
5. Si el equipo de trabajo considera fundamental la utilización de una herramienta CASE en el proceso de desarrollo de la aplicación, ¿Cuáles de las características considera fundamental que posea la herramienta?
  - a. Que esté integrado con las actividades de las metodologías de diseño.

- b. Que tenga controles de coherencia del diseño de la aplicación.
- c. Que maneje las versiones de los diseños realizados.
- d. Que genere parte de la interfaz y código de la aplicación.
- e. Que genere parte de la documentación de la aplicación.

6. El tiempo para la entrega de la aplicación es:

- a. Holgado
- b. Ajustado
- c. Muy Ajustado

7. ¿El equipo de programadores tiene experiencia en el uso lenguajes de programación para aplicaciones Web?

- a. Si
- b. No